

单片微型机

原理、应用与实验

(修订版)

● 张友德 赵志英 涂时亮 编



复旦大学出版社

单片微型机原理、应用与实验

(修 订 版)

张友德 赵志英 涂时亮 编



复旦大学出版社

JS143/18

内 容 简 介

本书是 1992 年版的修订本,基于各系列单片机的最新资料和开发应用的新成果,对原书的章节内容作了增删和调整,使其更具有系统性、先进性和实用性。

本书系统地论述了单片机的组成原理和单片机的程序设计、系统扩展、输入/输出设备的接口技术,并对单片机应用系统的硬件设计、可靠性和保密性设计、软件的结构设计作了专门的讨论,同时还介绍了典型的单片机应用系统的设计思想和实现方法,最后编排了单片机实验。各章都附有一定数量的习题和思考题。

本书可作为大专院校的单片机教材和科技人员的单片机应用技术参考书。

单片微型机原理、应用与实验

(修订版)

张友德 赵志英 涂时亮 编

出 版 复旦大学出版社

(上海国权路 579 号 邮政编码 200433)

发 行 新华书店上海发行所

照 排 南京理工大学激光照排公司

印 刷 复旦大学印刷厂

开 本 787×1092 1/16

印 张 25.25

字 数 624,000

版 次 1992 年 3 月第 1 版,1996 年 7 月第 2 版,1996 年 10 月第 8 次印刷

印 数 48 001—54 000

书 号 ISBN 7-309-01707-2/T·152

定 价 30.00 元

本版图书如有印订质量问题,请向承印厂调换。

修订版前言

单片微型计算机又名微控制器,是70年代中期发展起来的一种大规模集成电路器件。它在一块芯片内集成了计算机的各种功能部件,构成一种单片式的微型计算机。80年代以来,国际上单片机的发展很快,其产品之多令人目不暇接,出现了很多高性能的新型单片机,以适应各个领域的不同需要。

由于单片机具有功能强、体积小、可靠性高、面向控制和价格低廉等一系列优点,因而在工业、农业、国防、交通、民用消费品等各个领域得到了广泛的应用,有力地推动了各行业的技术改造和产品的更新换代。近几年来,国内单片机的开发应用也发展很快,取得了许多科研成果,并已转化为一定的生产力,收到了明显的经济和社会效益。

单片机的应用技术是一项新型的工程技术,广大科技人员希望能掌握它为四化建设服务,很多大专院校增设了单片机课程。本书作为单片机教材和科技人员的参考书,自1992年出版以来,得到广大读者的支持和帮助,收到了许多宝贵的意见和建议,我们表示衷心感谢!

根据近几年来教学和科研的实践以及新涌现的单片机性能,趁这次修订机会,对本书的部分章节作了适当的增删,并重新改编了第九~十二章。这样使本书的内容更具系统性、先进性和实用性。

纵观各个系列单片机特性,MCS-51系列单片机比较典型和流行,在单片机应用中占重要地位。因此,本书以MCS-51系列单片机为主展开讨论,但所论述的原理、方法同样适用于其它系列的单片机。

本书共分十二章,第一章为基础知识和单片机概论;第二~六章为MCS-51单片机原理部分,第七和八章为单片机的应用,第九~十二章为单片机实验指导。本书的第一~四章和第六~十二章由张友德编写,第五章由赵志英编写。涂时亮对全书各章提出了修改意见,最后由徐君毅和陈章龙审定。在编写过程中得到全国单片机学会和复旦大学计算机系微机室领导的关心和指导。薛剑虹为本书整理了大量资料和图片,杨胜球提供了实验资料,在出版过程中得到陆盛强的大力相助,在此谨向他们致谢!

由于时间和水平所限,书中错误和不妥之处难免,请读者批评指正。

编 者

1995年12月于上海

目 录

修订版前言

第一章 概 述

§ 1.1 计算机系统导论	1
§ 1.1.1 计算机的发展概况	1
§ 1.1.2 计算机的系统组成	2
§ 1.1.3 数制及其转换	3
§ 1.1.4 计算机中数的表示方法	6
§ 1.1.5 指令和指令系统	9
§ 1.1.6 存 贮 器	11
§ 1.1.7 运 算 器	12
§ 1.1.8 控 制 器	19
§ 1.1.9 外围设备	21
§ 1.1.10 总 线	21
§ 1.2 单片微型计算机	22
§ 1.2.1 单片机的发展历史	22
§ 1.2.2 单片机的发展趋势	23
§ 1.2.3 单片机内部结构	23
§ 1.2.4 单片机应用系统结构	25
§ 1.2.5 典型的单片机产品	26
§ 1.2.6 单片机的应用	34
习题和思考题	36

第二章 MCS-51 系列单片机系统结构

§ 2.1 总体结构	37
§ 2.2 中央处理器 CPU	39
§ 2.2.1 运 算 器	39
§ 2.2.2 控 制 器	39
§ 2.3 存贮器组织	44
§ 2.3.1 程序存贮器	44
§ 2.3.2 内部 RAM 数据存贮器	45
§ 2.3.3 特殊功能寄存器	46
§ 2.3.4 位存贮器	47
§ 2.3.5 外部 RAM 和 I/O 口	48
§ 2.4 I/O 口	48

§ 2.4.1 P1 口	19
§ 2.4.2 P3 口	49
§ 2.4.3 P2 口	50
§ 2.4.4 P0 口	50
§ 2.5 定时器/计数器	51
§ 2.5.1 定时器/计数器 T0 和 T1	51
§ 2.5.2 定时器/计数器 T2	55
§ 2.6 串行接口	57
§ 2.6.1 串行接口的组成和特性	58
§ 2.6.2 串行接口的工作方式	59
§ 2.6.3 波特率	66
§ 2.6.4 多机通信原理	68
§ 2.7 中断系统	69
§ 2.7.1 中断概念	69
§ 2.7.2 MCS-51 中断系统	70
§ 2.7.3 外部中断触发方式选择	74
§ 2.7.4 多外部中断源设计	75
§ 2.8 内部 EPROM 程序存储器	76
§ 2.8.1 8751H 的 EPROM	76
§ 2.8.2 8751BH 等的 EPROM	79
§ 2.9 CHMOS 型单片机的节电方式	82
§ 2.9.1 空闲方式	82
§ 2.9.2 掉电方式	83
§ 2.9.3 节电方式的应用	83
§ 2.10* 83C51FA/FB 的可编程计数器阵列(PCA)	85
§ 2.10.1 PCA 定时器/计数器	86
§ 2.10.2 比较/捕捉模块	87
§ 2.10.3 16 位捕捉方式	88
§ 2.10.4 软件定时器和高速输出方式	89
§ 2.10.5 脉冲宽度调制器方式	90
§ 2.10.6 监视定时器	91
§ 2.10.7 PCA 中的特殊功能寄存器地址	91
§ 2.11* 8×C51GA/GB 的 A/D 转换器	92
§ 2.11.1 A/D 特殊功能寄存器	92
§ 2.11.2 A/D 转换速度控制	93
§ 2.11.3 A/D 中断	93
习题和思考题	94

第三章 MCS-51 指令系统

§ 3.1 指令格式	95
§ 3.1.1 汇编指令	95
§ 3.1.2 伪指令	95

§ 3.1.3 常用的缩写符号	96
§ 3.2 寻址方式	97
§ 3.2.1 寄存器寻址	97
§ 3.2.2 直接寻址	98
§ 3.2.3 寄存器间接寻址	98
§ 3.2.4 立即寻址	99
§ 3.2.5 基寄存器加变址寄存器间接寻址	99
§ 3.3 数据操作和指令类型	100
§ 3.3.1 数据操作	100
§ 3.3.2 指令类型	101
§ 3.4 数据传送指令	102
§ 3.4.1 内部数据传送指令	102
§ 3.4.2 累加器 A 与外部数据存储器传送指令	106
§ 3.4.3 查表指令	106
§ 3.5 算术运算指令	107
§ 3.5.1 加法指令	107
§ 3.5.2 减法指令	110
§ 3.5.3 乘法指令	111
§ 3.5.4 除法指令	111
§ 3.6 逻辑运算指令	112
§ 3.6.1 累加器 A 的逻辑操作指令	112
§ 3.6.2 两个操作数的逻辑操作指令	113
§ 3.7 位操作指令	115
§ 3.7.1 位变量传送指令	115
§ 3.7.2 位变量修改指令	116
§ 3.7.3 位变量逻辑操作指令	116
§ 3.8 控制转移指令	117
§ 3.8.1 无条件转移指令	117
§ 3.8.2 条件转移指令	119
§ 3.8.3 调用和返回指令	121
§ 3.9 指令的应用	124
习题和思考题	126

第四章 汇编语言程序设计

§ 4.1 程序设计方法	128
§ 4.1.1 程序设计的步骤	128
§ 4.1.2 循环程序设计	128
§ 4.1.3 子程序设计和参数传递方法	134
§ 4.2 程序设计举例	140
§ 4.2.1 定点数运算程序	140
§ 4.2.2 查表程序	149

§ 4.2.3 数制转换程序	154
§ 4.2.4 输入/输出处理程序	157
习题和思考题	169

第五章 单片机系统扩展技术

§ 5.1 MCS-51 系统扩展原理	171
§ 5.2 程序存贮器的扩展	172
§ 5.2.1 常用的程序存贮器	172
§ 5.2.2 MCS-51 程序存贮器的扩展	179
§ 5.3 数据存贮器的扩展	182
§ 5.3.1 常用的数据存贮器	182
§ 5.3.2 MCS-51 扩展外部数据存贮器的原理	184
§ 5.3.3 典型的 MCS-51 存贮器扩展线路	185
§ 5.4 并行接口的扩展	186
§ 5.4.1 可编程并行接口 8255A	186
§ 5.4.2 用 74 系列器件扩展并行 I/O 口	194
§ 5.5 RAM/IO 扩展器 8155	195
§ 5.6* 串行接口的扩展	199
§ 5.6.1 可编程通用异步通信接口 8250	200
§ 5.6.2 RS-232C 总线	208
§ 5.7 D/A 接口的扩展	212
§ 5.7.1 梯形电阻式 D/A 转换原理	213
§ 5.7.2 DAC0832	214
§ 5.7.3 AD7520	217
§ 5.8 A/D 接口的扩展	218
§ 5.8.1 双积分 A/D 转换原理	219
§ 5.8.2 MC14433	220
§ 5.8.3 逐次逼近法 A/D 转换原理	223
§ 5.8.4 ADC0816	223
§ 5.8.5 ADC0809	226
习题和思考题	227

第六章 单片机的输入/输出设备接口

§ 6.1 拨码盘接口	229
§ 6.1.1 BCD 码拨盘的构造	229
§ 6.1.2 BCD 码拨盘的接口方法	230
§ 6.2 七段发光显示器接口	231
§ 6.2.1 显示器的结构	231
§ 6.2.2 显示器的工作方式和显示程序设计	232
§ 6.3 键盘接口	235
§ 6.3.1 键盘工作原理	235

§ 6.3.2 键盘接口方法	236
§ 6.3.3 键输入程序设计方法	236
§ 6.3.4 定时扫描方式	239
§ 6.3.5 中断扫描方式	239
§ 6.4 打印机接口	240
§ 6.4.1 PP40 的接口信号	240
§ 6.4.2 PP40 的操作方式	241
§ 6.4.3 PP40 的接口方法	244
§ 6.4.4 打印程序设计举例	244
习题和思考题	252

第七章 单片机应用系统研制方法

§ 7.1 单片机应用系统的设计	254
§ 7.1.1 单片机应用系统研制过程	254
§ 7.1.2 总体设计	254
§ 7.1.3 硬件设计	256
§ 7.1.4 可靠性设计	259
§ 7.1.5 保密性设计	263
§ 7.1.6 软件设计	264
§ 7.2 单片机开发系统	268
§ 7.2.1 单片机的开发与开发工具	268
§ 7.2.2 单片机开发系统的功能	270
§ 7.2.3 典型的单片机开发系统	272
§ 7.3 单片机应用系统调试	276
§ 7.3.1 硬件调试方法	276
§ 7.3.2 软件调试方法	278
习题和思考题	280

第八章 典型单片机应用系统设计举例——GPCR31 可编程顺序控制器

§ 8.1 顺序控制器的设计方法	281
§ 8.1.1 顺序控制的特点	281
§ 8.1.2 顺序控制器的设计方法	282
§ 8.1.3 顺序控制器系统的结构	283
§ 8.2 GPCR31 的硬件结构	285
§ 8.3 GPCR31 的功能	286
§ 8.4 软件设计思想	286
§ 8.4.1 定时控制算法	287
§ 8.4.2 条件控制算法	287
§ 8.5 软件框图	289
§ 8.5.1 编辑状态主程序	289
§ 8.5.2 运行状态主程序	291

§ 8.5.3 定时中断服务程序	291
------------------------	-----

第九章 EXR51 II 型单片机实验仪

§ 9.1 EICE51 的系统结构和功能	294
§ 9.1.1 EICE51 逻辑结构图	294
§ 9.1.2 EICE51 的功能	295
§ 9.1.3 系统连接和一般使用方法	296
§ 9.1.4 EICE51 的状态及其转换	297
§ 9.1.5 EICE51 仿真 RAM 的用途和使用方法	298
§ 9.2 EICE51 系统操作命令	298
§ 9.2.1 操作命令格式	298
§ 9.2.2 状态和方式转换命令	299
§ 9.2.3 信息传送命令	300
§ 9.2.4 读出检查命令	302
§ 9.2.5 读出/修改命令	303
§ 9.2.6 运行控制命令	305
§ 9.2.7 符号化调试命令	306
§ 9.3 FD-EDIT 行编辑使用方法	309
§ 9.3.1 一般使用方法	309
§ 9.3.2 行编辑命令	310
§ 9.3.3 编辑控制命令	312
§ 9.3.4 MCS-51 汇编命令	313
§ 9.4 FD-ASM51 扩展汇编使用方法	314
§ 9.4.1 符号及数据表示方法	314
§ 9.4.2 伪指令	316
§ 9.4.3 汇编指令格式	317
§ 9.4.4 汇编出错信息	318
§ 9.5 EICE51 和 IBM PC 机连接使用方法	319
§ 9.5.1 一般使用方法	319
§ 9.5.2 记盘操作	320
§ 9.5.3 目标程序记盘方法	320
§ 9.5.4 源程序的磁盘操作	321
§ 9.6 EICE51 键盘操作方法	321
§ 9.6.1 操作面板功能说明	321
§ 9.6.2 键盘操作方法	323

第十章 软件实验

§ 10.1 无符号十进制数加法实验	334
§ 10.2 无符号十进制数减法实验	335
§ 10.3 双字节 BCD 码乘法实验	336

§ 10.4	单字节 BCD 码除法实验	338
§ 10.5	电子钟实验(定时器、串行口、中断综合实验)	339
§ 10.6	定时器实验	341
§ 10.7	数据排序实验	342
§ 10.8	数据传送实验	343
§ 10.9	整数二翻十实验	344
§ 10.10	查表程序实验	345
§ 10.11	键盘实验	346

第十一章 接口实验

§ 11.1	可编程 I/O 接口 8255 实验	348
§ 11.2	TTL 输入/输出实验	351
§ 11.3	MC14433 A/D 转换实验	353
§ 11.4	ADC0809 A/D 转换实验	356
§ 11.5	DAC0832 D/A 转换实验	360

第十二章 应用实验

§ 12.1	步进电机控制实验	363
§ 12.2	直流电机转速测量与控制实验	368
§ 12.3	温度测量实验	373
§ 12.4	力测量实验	376
§ 12.5	智能显示屏实验	379

附录 1	EICE51 实验仿真器逻辑图	384
------	-----------------------	-----

附录 2	MCS-51 指令表	385
------	------------------	-----

参考书目	390
------------	-----

第一章 概 述

§ 1.1 计算机系统导论

§ 1.1.1 计算机的发展概况

一、计算机的发展历史

电子计算机是一种能够高速而精确地进行各种数据处理的机器,这是人类生产和科学技术发展的产物,它的出现又有力地推动了生产力的发展,目前计算机已应用到国民经济的各个领域,当代社会已离不开计算机。自从计算机诞生以来,它的发展经历了四代:

第一代计算机(40年代末期至50年代末期)是电子管计算机,所使用的元件主要是电子管。世界上第一台电子计算机是由美国宾夕法尼亚大学的 J. W. Mauchly 和 J. P. Eckert 研制成的 ENIAC 计算机,这台计算机用了 18 800 只电子管,加法速度是每秒 5 000 次,乘法速度是每秒 56 次。

第二代计算机(50年代末期至60年代末期)是晶体管计算机,所用的主要元件是晶体管。1957年,美国研制成了第一台晶体管计算机(TRANSACS-1000机)。

第三代计算机(60年代中期开始)是集成电路计算机,所用的元器件是小规模或中规模集成电路,如 IBM 公司于 1964 年推出的 IBM360 系列机。

第四代计算机(70年代初期开始)是大规模集成电路计算机。所使用的元件是大规模或超大规模集成电路。1971年 IBM370 系列机首先使用了大规模集成电路构成主存贮器。1975年研制成功了以大规模集成电路做主存贮器和逻辑元件的大型计算机,例如 470V/6 型 M-190 机等。

当前计算机的发展趋势是微型化、巨型化、网络化和智能模拟。

二、计算机的种类

计算机有模拟计算机和数字计算机两种。

模拟计算机(analog computer)是对模拟量进行操作的计算机,这种计算机解题速度快,但精度差。

数字计算机(digital computer)是对数据进行算术和逻辑操作的计算机,这种计算机在运算过程中全部自动化,具有运算速度快、精度高、通用性强等特点。

我们通常所说的电子计算机,实际上是指电子数字计算机。

电子计算机的种类很多。根据设计的目标来分,有通用计算机和专用计算机;根据用途来分,有科学计算、数据处理和工业控制计算机;若根据规模和功能来分,有巨型机、大型机、中型机、小型机、超小型机和微型机。

微型计算机又可以划分为多片微型机和单片微型计算机。

§ 1.1.2 计算机的系统组成

电子计算机是模仿人脑部分功能的一种工具,俗称为“电脑”。它的结构特点与工作过程和人脑有许多相似之处,电子计算机的工作原理模拟人手工计算的过程。试看一下人用算盘来计算 $(2436+3748-4569)$ 的过程,如果我们把算盘记为 R,则计算过程如表 1-1 所示。

表 1-1 使用算盘解题过程

序 号	操 作 命 令	注 释
0	$0 \rightarrow R$	清除算盘盘面
1	$2436 \rightarrow R$	在算盘上拨上 2436
2	$(R) + 3748 \rightarrow R$	在算盘中加上 3748
3	$(R) - 4569 \rightarrow R$	在算盘中减去 4569
4	(R)抄送纸上	抄送运算结果
5	停止	计算结束

在执行六步操作以后,在算盘上存放着运算结果 1615。表 1-1 是用算盘求解过程的形式描述,算盘 R 具有累加的作用。改变表中的操作命令,就可以实现其它问题的计算。

计算过程中的每一步都是一条指示人完成相应操作的命令,我们将这种执行某种操作的命令称为指令,完成某种功能的一组指令系列称之为程序。表 1-1 列出的六条指令就构成了一个程序,编制解题过程的过程称为程序设计。

在手工计算中,是由使用算盘的人按写在纸上的程序和数据来进行计算的。因此使用算盘的人、算盘、记录数据和程序的纸是计算过程中必不可少的组成部分。

若用电子计算机模拟上述解题过程,计算机必须具备下列条件:

- (1) 为了能进行各种数据运算,机器内必须有一个相当于算盘的运算器;
- (2) 为了保存和记录原始数据、解题程序和运算的中间结果,机器内必须有容量足够大的存贮器,这相当于手工计算时用的纸张;
- (3) 必须有按照解题程序指挥、控制各个部件协同工作的控制器,这相当于手工计算中的人脑;
- (4) 必须具备将原始数据和程序送入机器内部的输入设备和给出计算结果的输出设备;
- (5) 机器内应有必要的程序,以便开机后执行该程序,启动系统工作,自动地投入运行状态。

运算器、控制器、存贮器和输入/输出设备是计算机赖以工作的物质基础,称之为硬件,而运算器和控制器是计算机硬件的核心,称为中央处理器 CPU(Central Processing Unit)。

计算机的硬件结构如图 1-1 所示。

计算机中的程序称为软件。它包括使系统自动工作或提高计算机工作效率的系统软件和实现某一应用目标的应用软件。软件是计算机系统工作的“灵魂”。

计算机的工作也可以认为是信息加工过程。计算机中的信息是指数据或指令,它们是以一定的编码形式表示的,其意义各不相同,大致可分为:

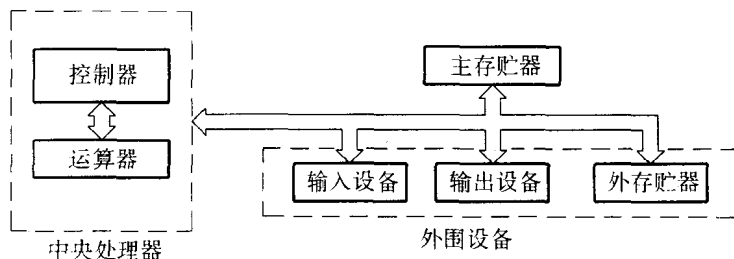
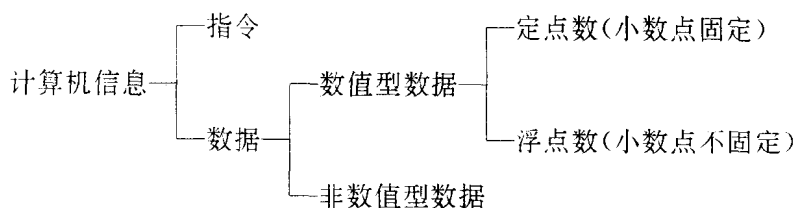


图 1-1 电子计算机硬件结构

§ 1.1.3 数制及其转换

一、进位计数制

进位计数制可概括如下：

● 有一个固定的基数 r , 数的每一位只能取 r 个不同的数字, 即符号集是 $\{0, 1, 2, \dots, r-1\}$;

● 逢 r 进位, 它的第 i 个数位对应于一个固定的值 r^i , r^i 称为该位的“权”。小数点左面各位的权是基数 r 的正次幂, 依次为 $0, 1, 2, \dots, m$ 次幂, 小数点右面各位的权是基数 r 的负次幂, 依次为 $-1, -2, \dots, -n$ 次幂。

以下我们用 $()_r$ 表示括号内的数是 r 进制数。将 r 进制数 $(a_m a_{m-1} \dots a_1 a_0 . a_{-1} a_{-2} \dots a_{-n})_r$ 按权展开, 表达式为:

$$a_m \times r^m + a_{m-1} \times r^{m-1} + \dots + a_1 \times r^1 + a_0 \times r^0 + a_{-1} \times r^{-1} + a_{-2} \times r^{-2} + \dots + a_{-n} r^{-n}$$

1. 十进制数

十进制数的基数 $r=10$, 符号集为 $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, 其权为: $\dots, 10^2, 10^1, 10^0, 10^{-1}, 10^{-2}, \dots$ 。

例 1 $(987.32)_{10} = 9 \times 10^2 + 8 \times 10^1 + 7 \times 10^0 + 3 \times 10^{-1} + 2 \times 10^{-2}$

2. 八进制数

八进制数的基数 $r=8$, 符号集为 $\{0, 1, 2, 3, 4, 5, 6, 7\}$, 其权为: $\dots, 8^2, 8^1, 8^0, 8^{-1}, 8^{-2}, \dots$ 。

例 2 $(7061.304)_8 = 7 \times 8^3 + 0 \times 8^2 + 6 \times 8^1 + 1 \times 8^0 + 3 \times 8^{-1} + 0 \times 8^{-2} + 4 \times 8^{-3}$

3. 十六进制数

十六进制数的基数 $r=16$, 符号集为 $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F\}$, 其权为: $\dots, 16^2, 16^1, 16^0, 16^{-1}, 16^{-2}, \dots$ 。

例 3 $(-A0.8F)_{16} = -(10 \times 16^1 + 0 \times 16^0 + 8 \times 16^{-1} + 15 \times 16^{-2})$

4. 二进制数

二进制数的基数 $r=2$, 符号集为 $\{0,1\}$, 权为 $\dots, 2^2, 2^1, 2^0, 2^{-1}, 2^{-2}, \dots$ 。

例 4 $(1011.101)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$

十进制、二进制、八进制和十六进制数码对照见表 1-2, 二进制与十进制小数对照见表 1-3。

表 1-2 十进制、二进制、八进制、十六进制数码对照表

十进制	二进制	八进制	十六进制	十进制	二进制	八进制	十六进制
0	0000	00	0	8	1000	10	8
1	0001	01	1	9	1001	11	9
2	0010	02	2	10	1010	12	A
3	0011	03	3	11	1011	13	B
4	0100	04	4	12	1100	14	C
5	0101	05	5	13	1101	15	D
6	0110	06	6	14	1110	16	E
7	0111	07	7	15	1111	17	F

表 1-3 二进制与十进制小数对照表

二进制小数	十进制小数	二进制小数	十进制小数
0.1	0.5	0.00001	0.03125
0.01	0.25	0.000001	0.015625
0.001	0.125	⋮	⋮
0.0001	0.0625		

二、二进制编码的十进制数

常用二进制编码的十进制数有 8421BCD 码(简称 BCD 码)、2421 码、5211 码和余 3 码等。它们都是用 4 位二进制数来表示 1 位十进制数。前三种码都是有码权, 余 3 码为无权码, 而 2421 码和 5211 码表示的十进制数不唯一, 8421BCD 码和余 3 码唯一表示一个十进制数。这四种编码的关系如表 1-4 所示。

表 1-4 四种编码的关系

8421BCD 码	2421 码	5211 码	余 3 码
0000	0000(或 0000)	0000(或 0000)	0011
0001	0001(或 0001)	0001(或 0010)	0100
0010	0010(或 1000)	0011(或 0100)	0101
0011	0011(或 1001)	0101(或 0110)	0110
0100	0100(或 1010)	0111(或 0111)	0111
0101	1011(或 0101)	1000(或 1000)	1000
0110	1100(或 0110)	1010(或 1001)	1001
0111	1101(或 0111)	1100(或 1011)	1010
1000	1110(或 1110)	1110(或 1101)	1011
1001	1111(或 1111)	1111(或 1111)	1100

三、进位计数制之间的转换

不同基的进位计数制之间数的转换, 一般有以下几种方法:

1. 直接相乘法

将表示成 r 进制数的 M 转换为 t 进制数。即基数 r 用基数 t 来表示, M 的各位数字用 t 进制的数系来表示, 然后作乘法和加法, 结果便是 t 进制数。

例 5 把十进制数 725 转换为二进制数。

$$\begin{aligned}(725)_{10} &= 7 \times 10^2 + 2 \times 10^1 + 5 \times 10^0 \\ &= 111 \times 1010^2 + 10 \times 1010^1 + 101 \times 1010^0 \\ &= (1011010101)_2\end{aligned}$$

2. 余数法(适合于整数部分转换)

将表示成 r 进制的整数 M 转换为 t 进制数的整数, 除以 t 取余法。

例 6 把十进制数 62 转换为 2 进制数。

$2 \overline{) 62} \cdots \cdots$	余数 = 0	↑ 低位 ↑ 高位
$2 \overline{) 31} \cdots \cdots$	余数 = 1	
$2 \overline{) 15} \cdots \cdots$	余数 = 1	
$2 \overline{) 7} \cdots \cdots$	余数 = 1	
$2 \overline{) 3} \cdots \cdots$	余数 = 1	
$1 \cdots \cdots$	余数 = 1	

结果: $(62)_{10} = (111110)_2$

3. 取整法(适用于小数部分转换)

将 r 进制数的小数转换为 t 进制的小数, 乘 t 取整法。

例 7 把十进制小数 0.375 转换为二进制数。

$0.375 \times 2 = 0.750 \cdots \cdots$	整数 = 0	↑ 高位 ↓ 低位
$0.75 \times 2 = 1.50 \cdots \cdots$	整数 = 1	
$0.50 \times 2 = 1.00$	整数 = 1	
$(0.375)_{10} = (0.011)_2$		

注意: 将 r 进制小数转换为 t 进制小数时, 有时会是无限循环小数, 这时可根据要求进行取舍。

4. 递归法(适合于计算机转换)

把 r 进制数 M 转换为 t 进制数, 其方法是拆成整数和小数两个部分, 然后把用递归算法产生的已转换成 t 进制数的整数和小数部分拼起来:

例 8 将十进制数 4827.625 转换为二进制数。

$$\begin{aligned}(4827)_{10} &= (((4 \times 10 + 8) \times 10 + 2) \times 10 + 7) \times 10^0 \\ &= ((100 \times 1010 + 1000) \times 1010 + 10 \times 1010) + 111 \\ &= (1001011011011)_2 \\ (0.625)_{10} &= (6 + (2 + 5 \times 10^{-1}) \times 10^{-1}) \times 10^{-1} \\ &= (110 + (10 + 101 \times 1010^{-1}) \times 1010^{-1}) \times 1010^{-1} \\ &\simeq (0.101)_2\end{aligned}$$

结果: $(4827.625)_{10} = (1001011011011)_2 + (0.101)_2$

$$=(1001011011011.101)_2$$

1.1.4 计算机中数的表示方法

一、真值和机器数

一个数是由符号和数值两部分组成的。例如：

$$N_1 = +1001010(+74)$$

$$N_2 = -1001010(-74)$$

在计算机中数的符号也是用二进制码表示的，一般正数的符号用“0”表示，负数的符号用“1”表示。例如：

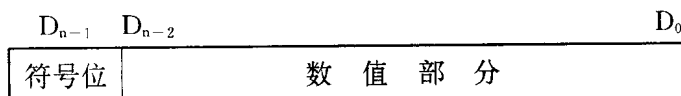
$$N_1 = 01001010(+74)$$

$$N_2 = 11001010(-74)$$

一个数在机器中的表示形式称为机器数，而把这个数本身称为真值。

二、带符号数的表示方法

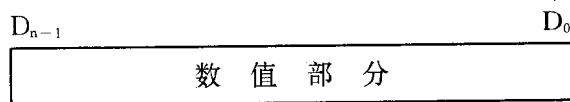
上面提到的机器数表示方法，用“0”表示正，用“1”表示负。这种表示数的方法，称为带符号数的表示方法。在机器中的一般表示形式为：



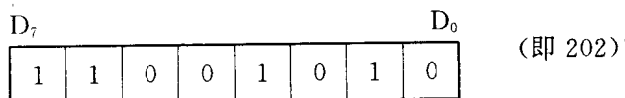
机器数最高位为符号位，其余的 $(n-1)$ 位为数值部分。

三、无符号数的表示方法

无符号数没有符号位，机器的全部有效位都用来表示数的大小。无符号数在机器中的一般形式为：



例如：



四、数的定点和浮点表示

十进制数 485.23 也可以表示为 0.48523×10^3 ，而在计算机内也有类似的两种数的表示方法，那就是定点数和浮点数。

1. 定点表示法

计算机内的定点数格式为：