

单片机原理及应用教程

21世纪高等院校计算机教材系列



单片机原理 及应用教程

- 刘瑞新 主编
- 赵全利 赵建军 翟红程 等编著

TP368.1

 机械工业出版社
CHINA MACHINE PRESS



21 世纪高等院校计算机教材系列

单片机原理及应用教程

刘瑞新 主编

赵全利 赵建军 翟红程 等编著



机械工业出版社

本书详尽地介绍了 MCS-51 系列单片机的硬件结构、指令系统及编程、内部功能及接口、系统外部功能的扩展、单片机系统开发及应用实例,最后介绍了新型串行总线接口芯片、其他单片机类型及选购。书中通过大量的例题和由浅入深的单片机应用实例,引导读者逐步认识、熟知、应用单片机。各章后配有习题以巩固学生所学的知识。

本书思路清晰,概念准确,层次结构分明,注重知识的内在联系与规律,便于自学。

本书可作为高等院校电类、机类及计算机类专业的教学用书,也可作为高职高专同类专业的教学用书及各类工程技术人员的自学用书。

图书在版编目(CIP)数据

单片机原理及应用教程/刘瑞新主编. —北京:机械工业出版社,2003.7

(21世纪高等院校计算机教材系列)

ISBN 7-111-12530-4

I. 单... II. 刘... III. 单片微型计算机—高等学校—教材 IV. TP368.1

中国版本图书馆CIP数据核字(2003)第054160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:胡毓坚

责任编辑:刘 青

责任印制:闫 焱

北京京丰印刷厂印刷·新华书店北京发行所发行

2003年7月第1版·第1次印刷

787mm×1092mm¹/₁₆·16.25印张·396千字

0 001—5 000册

定价:23.00元

凡购本图书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线电话(010)68993821、88379646

封面无防伪标均为盗版

出版说明

随着计算机技术的飞速发展，计算机在经济与社会发展中的地位日益重要。在高等院校的培养目标中，都将计算机知识与应用能力作为其重要的组成部分。根据计算机科学发展迅速的学科特点，计算机教育应面向社会，面向潮流，与社会接轨，与时代同行。随着计算机软硬件的不断更新换代，计算机教学内容也必须随之不断更新。

为满足高等院校计算机教材的需求，机械工业出版社聘请了清华大学、北方交通大学、北京邮电大学等院校的老师，经过反复研讨，结合当前计算机发展需要和编者长期从事计算机教学的经验，精心编写了这套“21世纪高等院校计算机教材”。

本套教材理论教学和实践教学相结合，图文并茂，内容实用，层次分明，讲解清晰，系统全面，其中溶入了老师大量的教学和科研经验，可用作各类高等院校、高等职业学校及相关院校的教材，也可作为培训班教材和自学用书。

前 言

随着计算机技术的高速发展，单片机以其自身的特点，已广泛应用于智能仪器、工业控制、家用电器、电子玩具等各个领域。这种将单片机嵌入到对象体系中的嵌入式系统已渗透到每个单位、每个家庭和个人。随着社会需求的不断增长，单片机的应用有着广泛的和稳定增长的市场。

多年来，以 8 位单片机组成的单片机应用系统，以其通用性强、价廉、设计灵活等特点而遍及各个领域。由于 8 位单片机综合性能的不不断提升，完全能够满足电子系统智能化及工业控制等很多应用领域不断提出的新要求，8 位单片机的应用仍呈繁荣之势。可以说，8 位单片机过去是、现在是、未来可预见的时期里也将是嵌入式应用的主流机型。

为了适应单片机应用发展的需要，培养具有单片机实际应用能力的高等专业人才，作者在总结多年单片机教学经验的基础上，大胆尝试单片机课程的教学改革后，编写了本教材。

本教材以单片机应用为主要目的，结合高等教育的特点，以当前仍处于主流地位且呈繁荣之势的 Intel 公司的 MCS-51 系列单片机为对象，介绍其内部资源、程序设计、系统扩展、人机界面、过程通道等应用开发中涉及到的问题，最后简单介绍了其他常用的单片机。

本书概念清楚，注重知识的内在联系与规律，采用归纳、类比的方法，系统地介绍了单片机的结构原理及应用系统的组成与设计方法，目的是使读者在较短的时间内掌握单片机的应用技术。为了便于读者理解、掌握本书的内容，每章均配有大量的例题与习题。

本教材从单片机应用开发的角度出发，对单片机教材的内容进行了整合，结构合理，面向设计；实例取材于工程实际，具有很高的参考价值；内容覆盖面广，条理清晰。不仅对学习掌握单片机，而且对实际开发也具有较好的帮助作用。本教材本着一切内容为应用服务的宗旨，在介绍单片机基本知识、基本结构、基本组成的基础上，结合各章的具体内容，通过大量由浅入深的单片机应用实例，引导读者逐步认识、熟知、应用单片机，力争达到触类旁通、举一反三的目的，为单片机的开发和深入应用打下坚实的基础。

本教材是依据中国计算机学会高等教育学会 2001 年审定的编写大纲编写的，所有例题和源程序都经上机调试通过，授课为 60 课时。

“单片机原理与应用”是高等学校计算机科学、自动控制等电类专业的一门专业必修课，随着单片机控制领域的不断拓展，机械、机电类专业也开设了此课程。本书可作为高等学校相关专业“单片机原理与应用”课程的教学用书，也可作为高职高专有关专业的教材，同时也可供从事单片机技术开发、应用的工程技术人员阅读、参考。

本书由刘瑞新主编，赵全利、赵建军、翟红程等编著，参与本书编写的还有肖兴达、王亚利、侯枫、胡健、李继方、郭新勇、袁红斌、马新起、张福莲、邓丹，全书由赵建军审校。

为了配合本书的教学，机械工业出版社为读者免费提供电子教案。可在机械工业出版社网站（www.cmpbook.com）上下载。

作 者

目 录

出版说明

前言

第 1 章 概述	1
1.1 数制与编码	1
1.1.1 数制及其转换	1
1.1.2 编码	4
1.2 单片机与嵌入式系统组成	7
1.2.1 单片机及其发展概况	7
1.2.2 单片机的特点和应用	8
1.2.3 嵌入式系统	9
1.2.4 单片机应用系统的结构	10
1.3 一个简单的单片机应用实例	11
1.4 习题	12
第 2 章 MCS-51 的结构	13
2.1 MCS-51 单片机总体结构	13
2.1.1 MCS-51 单片机总体结构框图及功能	13
2.1.2 MCS-51 引脚功能	16
2.2 MCS-51 存储结构及位处理器	19
2.2.1 MCS-51 存储器的特点	19
2.2.2 程序存储器	20
2.2.3 数据存储器	21
2.2.4 专用寄存器区 (SFR)	23
2.2.5 位处理器	25
2.3 MCS-51 工作方式和时序	25
2.3.1 复位方式	25
2.3.2 程序执行工作方式	26
2.3.3 节电工作方式	27
2.4 MCS-51 单片机的时序	28
2.4.1 时钟	28
2.4.2 CPU 时序	28
2.5 习题	32
第 3 章 MCS-51 单片机指令系统及编程举例	33
3.1 指令系统简介	33
3.1.1 指令格式	33
3.1.2 指令分类及符号说明	34

3.2 寻址方式	35
3.3 指令系统	39
3.3.1 数据传送类指令	39
3.3.2 算术运算类指令	45
3.3.3 逻辑运算类指令	48
3.3.4 控制转移类指令	51
3.3.5 位操作类指令	56
3.4 汇编语言程序设计基础	58
3.4.1 汇编语言程序的组成	59
3.4.2 伪指令	60
3.4.3 汇编语言程序设计实例	62
3.5 习题	69
第4章 MCS-51 单片机的中断系统	71
4.1 中断的概念	71
4.2 8051 中断系统结构及中断控制	72
4.2.1 中断源和中断请求标志	72
4.2.2 中断允许控制	74
4.2.3 中断优先级控制	74
4.2.4 中断响应过程	75
4.2.5 中断响应时间	76
4.2.6 中断响应后中断请求的撤除	76
4.3 中断系统应用实例	76
4.3.1 单步操作的中断实现	76
4.3.2 外部中断源扩展	77
4.4 中断程序举例	78
4.5 习题	82
第5章 MCS-51 单片机内部定时器/计数器	83
5.1 定时器/计数器简介	83
5.2 定时器/计数器的控制字	84
5.2.1 定时器/计数器的工作模式寄存器 TMOD	84
5.2.2 定时器控制寄存器 TCON	85
5.3 定时器/计数器工作模式	85
5.3.1 工作模式 0	85
5.3.2 工作模式 1	86
5.3.3 工作模式 2	86
5.3.4 工作模式 3	87
5.4 定时器/计数器应用实例	88
5.4.1 模式 0 的应用	88
5.4.2 模式 1 的应用	89

5.4.3	模式 2 的应用	89
5.4.4	模式 3 的应用	90
5.4.5	定时器/计数器综合应用	91
5.5	习题	92
第 6 章	单片机串行口及应用	94
6.1	MCS-51 单片机串行接口	94
6.1.1	串行通信的基本概念	94
6.1.2	串行口结构	97
6.1.3	串行口控制	98
6.2	串行口工作方式及波特率设置	99
6.2.1	串行口的工作方式	99
6.2.2	波特率设置	100
6.3	串行口应用及实例	102
6.3.1	串行口方式 0 的应用	102
6.3.2	串行口在其他方式下的应用	104
6.3.3	多机通信	106
6.4	习题	112
第 7 章	单片机的系统扩展	114
7.1	MCS-51 单片机最小系统	114
7.1.1	8051/8751 单片机最小系统	114
7.1.2	8031 最小应用系统	114
7.2	扩展总线的产生	115
7.3	程序存储器的扩展	116
7.3.1	常用的程序存储器	116
7.3.2	程序存储器的扩展	120
7.4	数据存储器的扩展	122
7.4.1	常用的数据存储器	122
7.4.2	数据存储器的扩展	123
7.4.3	扩展举例	124
7.5	I/O 端口的扩展	125
7.5.1	简单并行 I/O 口的扩展	126
7.5.2	8155 可编程多功能接口的扩展	128
7.6	习题	134
第 8 章	单片机系统的接口	135
8.1	键盘及接口电路	135
8.1.1	键盘的分类	135
8.1.2	键盘的工作原理	136
8.1.3	键盘接口扩展设计	138
8.2	显示及显示器接口	140

8.2.1	LED 状态显示	140
8.2.2	LED 数码显示	142
8.2.3	七段 LED 数码管显示接口	143
8.2.4	键盘、显示器综合设计	145
8.2.5	液晶显示器 (LCD) 在单片机中的应用	148
8.3	A/D、D/A 转换器与单片机的接口	154
8.3.1	D/A 转换器	154
8.3.2	A/D 转换器	157
8.4	应用举例	161
8.4.1	简易 5V 直流数字电压表	161
8.4.2	流水灯控制器	163
8.5	习题	164
第 9 章	单片机应用系统举例与开发	166
9.1	单片机应用系统举例	166
9.1.1	光电计数器	166
9.1.2	时间顺序控制器	167
9.1.3	转速表	169
9.1.4	交通灯管理系统	170
9.2	单片机系统的开发应用过程	173
9.2.1	总体论证	174
9.2.2	总体设计	175
9.2.3	硬件设计	175
9.2.4	软件设计	175
9.2.5	联机调试	176
9.2.6	脱机运行	177
9.3	单片机开发系统	177
9.3.1	开发系统的基本功能	177
9.3.2	开发系统简介	177
9.4	单片微机系统开发应用举例	180
9.4.1	系统简介	180
9.4.2	研制过程介绍	181
9.5	习题	189
第 10 章	新型串行总线接口芯片	191
10.1	I ² C 接口芯片	191
10.1.1	I ² C 总线简介	191
10.1.2	AT24Cxx 芯片与 MCS-51 的连接	195
10.1.3	通用 I ² C 总线发送/接收程序	203
10.1.4	PCF8563 时钟/日历芯片与 MCS-51 的连接	208
10.2	SPI 接口芯片与 MCS-51 的连接	211

10.2.1	SPI 总线简介	211
10.2.2	X25043 与 MCS-51 的连接	213
10.2.3	MC14489 与 MCS-51 的连接	218
10.3	习题	221
第 11 章	其他单片机简介及选购	222
11.1	80C51 系列单片机	222
11.2	89CXX 单片机	223
11.3	单片机的选购	224
11.4	优秀单片机产品推荐	226
11.4.1	MSP430 系列的 16 位单片机	226
11.4.2	高速 8051 片上系统单片机 C8051F300	227
11.4.3	ST 公司的 ST62 系列单片机	228
11.4.4	在系统可编程的 8 位低压微控制器 AT89S8252	228
11.4.5	ADI 公司的微转换器 AD μ C812	229
11.4.6	“嵌入式高速”可编程系统器件 μ PSD3000 系列	229
附录	231
附录 A	MCS-51 指令表	231
附录 B	ASCII (美国标准信息交换码) 码表	235
附录 C	典型汇编及模拟运行程序的使用	236
C.1	源程序汇编	236
C.2	源程序的模拟运行	241

第1章 概 述

本章主要讲述计算机的基础知识，包括：常用的数制及其不同数制相互间的转换；二进制数的编码、二一十进制编码、ASCII 码；单片机的概念、应用领域及其嵌入式系统概念。最后通过一个简单的单片机应用实例使读者初步了解单片机的工作概况。

1.1 数制与编码

在计算机中，任何命令和信息都是以二进制数据的形式存储的。计算机所执行的全部操作都归结为对数据的处理和加工，为了便于理解单片机系统的基本工作原理，掌握数字、字母等字符在单片机系统中的表示方法及处理过程，本章首先介绍有关数制和编码等方面的基础知识。

1.1.1 数制及其转换

数制就是计数方式。

日常生活中常用的是十进制计数方式，即逢十进一。在计算机中，由于所采用的电子逻辑器件的特点，计算机内部一切信息存储、处理和传送均采用二进制数的形式。可以说，二进制数是计算机能直接识别并进行处理的惟一形式。但是，在向计算机输入数据及输出数据时，人们惯于用十进制、十六进制数据等，因此，计算机在处理数据时，必须进行各种数制之间的相互转换。

1. 二进制数

二进制数只有两个数字符号：0 和 1。计数时按“逢二进一”的原则进行计数。也称其基数为二。

根据位权表示法，每一位二进制数在其不同位置表示不同的值。例如：

1	1+1=10	1+1+1+1=100	1+1+1+1+1+1+1+1=1000
↓	↓	↓	↓
1 (即 2^0)	2 (即 2^1)	4 (即 2^2)	8 (即 2^3)

对于任何二进制数，可按位权求和展开，则有

$$(10)_2 = 1 \times 2^1 + 0 \times 2^0 = (2)_{10}$$

$$(11)_2 = 1 \times 2^1 + 1 \times 2^0 = (3)_{10}$$

$$(110)_2 = 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = (6)_{10}$$

$$(111)_2 = 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = (7)_{10}$$

$$(1111)_2 = 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = (15)_{10}$$

$$(10110)_2 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = (22)_{10}$$

对于 8 位二进制数 (D0~D7)，各位所对应的值为

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
D7	D6	D5	D4	D3	D2	D1	D0

例如，二进制数 10110111，按位权展开求和计算可得

$$\begin{aligned}
 (10110111)_2 &= 1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \\
 &= 128 + 0 + 32 + 16 + 0 + 4 + 2 + 1 \\
 &= (183)_{10}
 \end{aligned}$$

对于含有小数的二进制数，小数点右边第一位小数开始向右各位的权值分别为

2^{-1}	2^{-2}	2^{-3}	2^{-4}	...

在计算机中，二进制数的每一位是数据的最小存储单位。将 8 位 (bit) 二进制数称为一个字节 (B)，字节是计算机存储信息的基本数据单位。存储器的容量常用以下单位表示：

$$1\text{B}=8\text{bit}$$

$$1024\text{B}=1\text{KB}$$

$$1024\text{KB}=1\text{MB}$$

$$1024\text{MB}=1\text{GB}$$

$$1024\text{GB}=1\text{TB}$$

若存储器内存容量为 64MB，即表示其容量为

$$\begin{aligned}
 64\text{MB} &= 64 \times 1024\text{KB} \\
 &= 64 \times 1024 \times 1024\text{B}
 \end{aligned}$$

在 MCS-51 单片机中，存储器容量一般可扩展为 64KB。

必须指出：在计算机中，一个二进制数（如 8 位、16 位或 32 位）既可以表示数值，也可以表示一种符号，也可以表示某种操作（即指令），计算机在程序运行时自动识别。这就是本节开始所述及的，一切信息都是以二进制数据进行存储的。

2. 十六进制数

计算机在输入输出或书写时，可采用十六进制数表示相应的二进制数。

十六进制数有 16 个数字符号，其中 0~9 与十进制相同，剩余 6 个为 A~F，分别表示十进制数的 10~15，计数原则是逢“十六进一”，也称其基数为十六。例如：

$$(31)_{16} = 3 \times 16^1 + 1 \times 16^0 = (49)_{10}$$

$$(2AF)_{16} = 2 \times 16^2 + 10 \times 16^1 + 15 \times 16^0 = (687)_{10}$$

为了便于区别不同进制的数据，一般情况下可在数据后跟一后缀：

二进制数用“B”表示（如 00111010B）；

十六进制数用“H”表示（如 3AH）；

十进制数用“D”表示（如 39D 或 39）。

3. 不同数制之间的转换

前已述及，计算机中的数只能用二进制表示，十六进制数适合读写方便的需要，日常生活中使用的是十进制数，计算机根据需要对各种进制数据进行转换。

(1) 二进制数转换为十进制数

对任意二进制数均可按权值展开将其转化为十进制数。例如：

$$10111\text{B} = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 23\text{D}$$

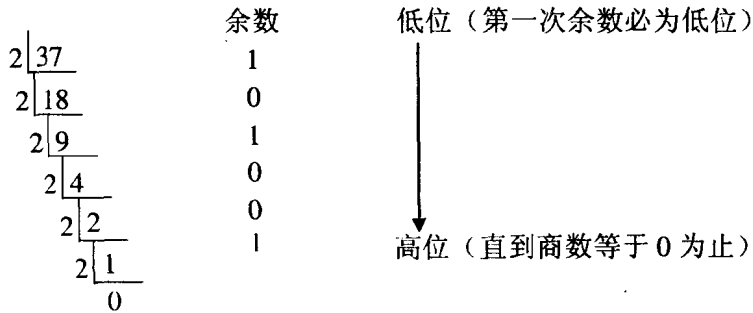
$$10111.011\text{B}=1\times 2^4+0\times 2^3+1\times 2^2+1\times 2^1+1\times 2^0+0\times 2^{-1}+1\times 2^{-2}+1\times 2^{-3}$$

$$=23.375\text{D}$$

(2) 十进制数转换为二进制数

十进制数转换为二进制数，可将整数部分和小数部分分别进行转换，然后合并。其中整数部分可采用“除2取余法”进行转换，小数部分可采用“乘2取整法”进行转换。

例如：采用“除2取余法”将37D转换为二进制数。



结果为 37D=100101B。

也可将十进制数与二进制位权从高位到低位比较，若十进制数大于或等于二进制某位，则该位取“1”，否则该位取“0”，采用按位分割法进行转换。

例如：将 37.625 转换为二进制数。

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1
0	0	1	0	0	1	0	1

将整数部分 37 与二进制各位权值从高位→低位比较， $37 > 32$ ，则该位取 1，剩余 $37 - 32 = 5$ ，逐位比较，得 00100101B。

将小数部分 0.625 按同样方法，得 0.101B。

结果为 37.625D=100101.101B。

(3) 二进制数与十六进制数的相互转换

在计算机进行输入、输出时，常采用十六进制数。十六进制数可看作是二进制数的简化表示。

因为 $2^4=16$ ，所以 4 位二进制数相当于 1 位十六进制数，其二进制、十进制、十六进制对应数的转换关系见表 1-1。

表 1-1 二进制、十进制、十六进制对应转换表

十 进 制	二 进 制	十 六 进 制
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6

(续)

十进制	二进制	十六进制
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

在将二进制数转换为十六进制数时，其整数部分可由小数点开始向左每4位为一组进行分组，直至高位。若高位不足4位，则补0使其成为4位二进制数，然后按表1-1对应关系进行转换。其小数部分由小数点向右每4位为一组进行分组，不足4位则末位补0使其成为4位二进制数，然后按表1-1对应关系进行转换。

例如：1000101B=0100 0101B=45H

10001010B=1000 1010B=8AH

100101.101B=0010 0101.1010B=25.AH

需要将十六进制数转换为二进制数时，则为上述方法的逆过程。

例如：45.AH=0100 0101. 1010 B

┌───┐ ┌───┐ ┌───┐
4 5 A

即 45.AH=1000101.101B

例如：7ABFH=0111 1010 1011 1111 B

┌───┐ ┌───┐ ┌───┐ ┌───┐
7 A B F

即 7ABFH =11110101011111B

(4) 十进制数与十六进制数的相互转换

十进制数与十六进制数的相互转换可直接进行，也可先转换为二进制数，然后再把二进制数转换为十六进制数或十进制数。

例如：将十进制数 37D 转为十六进制数。

37D=100101B=25H

例如：将十六进制数 41H 转换为十进制数。

41H=01000001B=65D

也可按位权展开求和方式将十六进制数直接转换为十进制数，这里不再详述。

1.1.2 编码

计算机内部所有数据均用二进制代码的形式表示，前面所提到的二进制数，没有涉及到正、负符号问题，实际上是一种无符号数的表示。但是在计算机中，数却有正、负之分。计

计算机通过输入设备（如键盘）输入信息和通过输出设备输出信息也是多种形式的，既有数字、字母，也有各种控制符号及汉字等。为此，需要对常用的数据及符号等进行编码，以表示不同形式的信息。这种以编码形式所表示的信息既便于存储，也便于由输入设备输入信息、输出设备输出相应的信息。

1. 二进制数的编码

(1) 机器数与真值

一个数在计算机中的表示形式叫做机器数。而这个数本身（含符号“+”或“-”）称为机器数的真值。

通常在机器数中，用最高位“1”表示负数，“0”表示正数（以下均以 8 位二进制数为例）。

例如，两个数 N1、N2 的真值为

$$N1=105=+01101001B$$

$$N2=-105=-01101001B$$

则对应的机器数为

$$N1=0\ 1101001B\ (\text{最高位“0”表示正数})$$

$$N2=1\ 1101001B\ (\text{最高位“1”表示负数})$$

(2) 原码、反码和补码

1) 原码。按上所述，正数的符号位用“0”表示，负数的符号位用“1”表示，其数值部分随后表示，称为原码。

仍以上面 N1、N2 为例，则

$$[N1]_{\text{原}}=0\ 1101001B$$

$$[N2]_{\text{原}}=1\ 1101001B$$

原码表示方法简单，便于与真值进行转换。但在进行减法时，为了把减法运算转换为加法运算（计算机结构决定了加法运算），必须引进反码和补码。

2) 反码、补码。在计算机中，任何有符号数都是以补码形式存储的。

对于正数，其反码、补码与原码相同。例如：

$$N1=+105$$

$$\text{则}[N1]_{\text{原}}=[N1]_{\text{补}}=[N1]_{\text{反}}=0\ 1101001B$$

对于负数，其反码为：原码的符号位不变，其数值部分按位取反。例如：

$$N2=-105$$

$$\text{则}[N2]_{\text{原}}=1\ 1101001B$$

$$[N2]_{\text{反}}=1\ 0010110B$$

负数的补码为：原码的符号位不变，其数值部分按位取反后再加 1（即负数的反码+1），称为求补。例如：

$$N2=-105$$

$$\text{则}[N2]_{\text{补}}=[N2]_{\text{反}}+1$$

$$=1\ 0010110+1=1\ 0010111B$$

如果已知一个负数的补码，可以对该补码再进行求补码（即一个数的补码的补码），即可得到该数的原码，即 $[[X]_{\text{补}}]_{\text{补}}=[X]_{\text{原}}$ ，而求出真值。例如：

已知: $[N2]_{补}=1\ 0010111B$

$$[N2]_{原}=11101000B+1=11101001B$$

可得真值: $N2=-105$

对采用补码形式表示的数据进行运算时, 可以将减法转换为加法。例如:

设 $X=10$, $Y=20$, 求 $X-Y$ 。

$X-Y$ 可表示为 $X+(-Y)$, 即 $10+(-20)$ 。

$$[X]_{原}=[X]_{反}=[X]_{补}=00001010B$$

$$[-Y]_{原}=10010100B$$

$$[-Y]_{补}=[-Y]_{反}+1=11101011+1=11101100B$$

则有 $[X+(-Y)]_{补}=[X]_{补}+[-Y]_{补}$

$$=00001010B+11101100B \text{ (按二进制相加)}$$

$$=11110110B \text{ (和的补码)}$$

再对 $[X+(-Y)]_{补}$ 求补码, 得

$$[X+(-Y)]_{原}=10001001+1=10001010B$$

则 $X-Y$ 的真值为 $-10D$ 。

必须指出: 所有负数在计算机中都是以补码形式存放的。对于 8 位二进制数, 作为补码形式, 它所表示的范围为 $-128 \sim +127$; 而作为无符号数, 它所表示的范围为 $0 \sim 255$ 。对于 16 位二进制数, 作为补码形式, 它所表示的范围为 $-32\ 768 \sim +32\ 767$; 而作为无符号数, 它所表示的范围为 $0 \sim 65\ 536$ 。

2. 二一十进制编码

二一十进制编码又称 BCD 编码, 既具有二进制数的形式, 以便于存储, 又具有十进制数的特点, 以便于进行运算和显示结果。在 BCD 码中, 用 4 位二进制代码表示 1 位十进制数。常用的 8421BCD 码的对应编码见表 1-2。

表 1-2 二一十进制编码 (8421BCD 码)

十进制数	8421BCD 码
0	0000B (0H)
1	0001B (1H)
2	0010B (2H)
3	0011B (3H)
4	0100B (4H)
5	0101B (5H)
6	0110B (6H)
7	0111B (7H)
8	1000B (8H)
9	1001B (9H)

例如, 将 27 转换为 8421BCD 码:

$$27D=(0010\ 0111)_{8421BCD \text{ 码}}$$

将 105 转换为 8421BCD 码:

105D=(0001 0000 0101)_{8421BCD}码

必须指出：8421BCD 码中不允许出现 1010~1111 这 6 个代码（因其值大于 9），称其为非法 8421BCD 码。

3. ASCII 码

对于计算机中非数值型数据，如

十进制数字符号：“0”，“1”，…，“9”（不是指数值）；

26 个大小写英文字母；

键盘专用符号：“#”、“\$”、“&”、“+”、“=”；

键盘控制符号：“CR”（回车）、“DEL”等。

上述这些符号在由键盘输入时不能直接装入计算机，必须将其转换为特定的二进制代码（即将其编码），以二进制代码所表示的字符数据的形式装入计算机。

ASCII（American Standard Code for Information）码是一种国际标准信息交换码，它利用 7 位二进制代码来表示字符，再加上 1 位校验位，故在计算机中用 1 个字节 8 位二进制数来表示一个字符，这样有利于对这些数据进行处理及传输。

常用字符的 ASCII 码表示见表 1-3。

表 1-3 常用字符的 ASCII 码

字 符	ASCII 码
0	00110000B (30H)
1	00110001B (31H)
2	00110010B (32H)
⋮	⋮
9	00111001B (39H)
A	01000001B (41H)
B	01000010B (42H)
C	01000011B (43H)
⋮	⋮
a	01100001B (61H)
b	01100010B (62H)
c	01100011B (63H)
⋮	⋮
CR (回车)	00001101B (0DH)

ASCII（美国标准信息交换码）码表见附录 B。

1.2 单片机与嵌入式系统组成

1.2.1 单片机及其发展概况

1. 什么是单片机

单片机（Single-Chip-Microcomputer）又称单片微控制器，其基本结构是将微型计算机