

微机原理及应用问答

孙志馨 尹更生 编



机械工业出版社

本书主要内容包括微机的数据传送、逻辑运算与逻辑操作、分支程序和循环程序、微机中的数值计算、子程序及其调用、汇编语言、输入/输出及简单接口、中断控制及微机在控制系统中的应用等。为使读者巩固所学的知识,书中还有较多的例题及练习。本书编写采用问答方式,精炼地叙述了微机基本知识和应用实例,同时还配备了一定数量的习题和实验。

本书适用于各类工程技术人员阅读,也可作为有关电类大专学生的参考书。

微机原理及应用问答

孙志馨 尹更生 编

*

责任编辑:孙 瑞

封面设计:郭景云

*

机械工业出版社出版(北京阜成门外百万庄南街一号)

(北京市书刊出版业营业许可证出字第117号)

国防工业出版社印刷厂印刷

新华书店北京发行所发行·新华书店经售

*

开本 $787 \times 1092 \frac{1}{32}$ · 印张 $13 \frac{3}{8}$ · 字数 292 千字
1987年9月北京第一版·1987年9月北京第一次印刷

印数 0,001—6,700 · 定价, 2.90元

*

统一书号: 15033·6873

前 言

微型计算机虽然只有十多年的历史，但其发展速度十分惊人，应用范围愈来愈广。在微机热浪推动下，很多人都想学习微机、掌握微机和应用微机。

微机为什么具有如此强大的吸引力呢？在学完本书后，你将感受到，它的出现打破了近百年来工业控制的传统方法。它不仅可在工业控制、而且还可在企业管理、小型计算、教学、家庭生活电气化等方面得到应用。除此之外，微机的价格也十分低廉。

一提到微机，常常会有一种神秘感，不知它有多么深奥，想学而不知从何入门，想用不知从何下手，万事开头难。

如果能有一把打开微机大门的钥匙，它不但能引导你入门，而且还指导你应用。那么，你一定会下决心去寻找这把钥匙的。

本书将试图为您提供这把钥匙。

* * *

微机芯片的种类很多，目前比较流行的是：

MOTOROLA 6800 系列

INTEL 8080 系列

ZILOG Z80 系列

这三种都是八位字长的微机芯片。

近几年来，这三家公司相继生产出 16 位字长的微芯片机，它们定名为：

M68000 系列

8086 系列

Z8000 系列

16 位字长的微机大有取代小型计算机的趋势。但是,从目 前生产和应用来看, 8 位字长的微机还占主导地位。

在我国,把这些微机芯片又组装成了各种型号的微型机,如Z80—STARTER、TK—80、SDK—85、TP—80 等等。有些厂家除生产单板微型机外,还把微机芯片与外部设备(如显示器 CRT、打印机、磁盘、A/D、D/A 等)相联接,构成一个微机系统。

* * * *

微机在我国已开始推广和应用,它所带来的经济效益使人倍受鼓舞,这就促使人们向深度和广度进一步探索它的应用。越来越多的有识之士已开始觉察到,微机可能会引起一场技术变革。打开报纸,我们经常会看到一些鼓舞人心的报道——微机之花开遍各条战线;交通业插上微机的翅膀;千帆竞用微机;电力工业应用微机自动调度等等。这些成就仅仅是个开端,更丰硕的成果等待人们去栽培、浇灌和收获。

* * * *

在学习微机前,还要提醒你注意学习微机的方法。无目的地泛学,不如有针对性地学。各种微机虽然基本原理相同,但它们的指令系统、接口芯片等都有差异。在实际使用中,人们往往根据现有条件、传统习惯,选用一种微机。因此,泛学不如专学,学会一种,定能举一反三。由于这个理由,在这本教材中,主要选用国内较流行的Z80 微机做范例,作为学习微机的入门。在必要的地方,也简略地介绍一点 8085 微机。

宋代学者朱熹说过，读书要三到，即“心到、眼到、口到”。在科学技术发达的今天，还应再加上“手到”。学习微机，如果不动手去实践，仅从书本上学，此法恐怕是不足取的。

这本书遵循上述两个指导思想而编写。因此，书中除采用问答方式精炼地讲解微机基本知识和应用实例外，还配备了一定数量的习题和实验。

如果有条件的话，希望你能配备一台 Z80 单板机，边学边做，这将会起到事半功倍之效果。

在编写这本读物过程中，得到了一些同志给予我们热情的帮助和支持，并提出了许多宝贵意见。在此，我们表示衷心的感谢。

由于我们水平有限，书中错误和不妥之处，在所难免，殷切希望读者给予批评指正。

作者 1985年
于西安交通大学

目 录

前言

第一章 微机的基本概念 1

1. 在应用微机前, 为什么要掌握二进制数和十六进制数? 1
2. 三种数制之间怎样转换? 4
3. 微机由几部分组成? 8
4. 什么叫做 CPU? 它有哪些外部功能? 9
5. CPU 的内部概貌怎样? 14
6. 什么叫做随机存贮器? 它与 CPU 怎样连接? 19
7. 只读存贮器 ROM 有什么特点? 29
8. 怎样阅读微机电路图? 33
9. 什么叫做指令、程序和助记符? 35

第二章 微机的数据传送 42

10. 微机中有哪些地方的数据需要相互传送? 42
11. 八位数据有几种传送方式? 42
12. 什么叫做寻址方式? 57
13. 十六位数据有几种传送方式? 61
14. 怎样成组地传送数据? 65
15. 怎样查阅指令表? 71
- 习题与实验 74

第三章 逻辑运算与逻辑操作 75

16. 基本的逻辑运算有几种? 75
17. 什么叫做“加 1”、“减 1”操作? 80
18. 怎样比较两个数的大小? 82
19. 在数据块中怎样检索“关键字”? 86

VIII

20. 微机怎样做移位操作?	91
21. 怎样使字节中某些位置“1”或置“0”?	98
习题与实验	100

第四章 分支程序和循环程序

22. 什么叫做分支程序和循环程序?	102
23. 为什么要画程序流程图?	104
24. 源程序中为什么要用“符号地址”代替存储器的 “真实地址”?	106
25. 什么叫做无条件转移?	107
26. 怎样使用 Z 标志构成循环程序或分支程序?	108
27. 怎样使用进位标志 Cy 构成循环或分支程序?	111
28. 怎样使用符号标志 S 构成循环或分支程序?	113
29. 怎样使用奇偶/溢出标志 P/V 构成循环或分支程序?	118
30. 循环程序和分支程序设计的规律是什么?	124
习题与实验	129

第五章 微机中的数值计算

31. 怎样进行二进制数的加减运算?	131
32. 怎样进行十进制数的加减运算?	136
33. 二进制数怎样转换成十进制数?	139
34. 怎样设计乘法运算程序?	142
35. 怎样设计除法运算程序?	145
36. 怎样设计求平方根程序?	147
37. 怎样求取函数值?	150
38. 怎样做积分运算?	155
39. 怎样做微分运算?	157
习题与实验	158

第六章 子程序及其调用

40. 什么叫做子程序?	160
41. 什么是“堆栈”? 它的特点和用途是什么?	161
42. 怎样调用子程序? 调用的形式有哪些?	169

43. 什么叫做“条件调用”和“条件返回”?	177
44. 怎样使用通用子程序?	178
45. 怎样编制子程序?	185
46. 怎样改变子程序的返回地址?	189
习题与实验	193

第七章 汇编语言 195

47. 为什么要使用汇编语言?	195
48. 汇编语句的格式怎样?	199
49. 什么叫做“伪指令”?	202
50. 伪指令ORG 和 END 的含意是什么?	203
51. 标号赋值伪指令 EQU 和 DL 的用途是什么?	205
52. 伪指令 DB 的用途是什么?	207
53. 伪指令 DW 的用途是什么?	210
54. 伪指令 DS 的用途是什么?	214
习题与实验	217

第八章 输入/输出及简单接口 219

55. 怎样用输入/输出指令实现输入/输出操作?	219
56. 怎样确定外设的地址及相应的译码电路?	223
57. 什么叫做直接输入/输出方式?	227
58. 什么叫做查询等待输入/输出方式?	233
习题与实验	238

第九章 中断控制 239

59. 什么叫做中断控制方式?	239
60. 哪些外设可使用中断控制方式进行输入/输出?	241
61. 中断控制方式需解决哪些至关重要的问题?	243
62. 中断控制方式中使用了哪些新指令?	247
63. 怎样编写中断控制程序?	254
64. CPU 怎样区别中断源的重要程度?	258
65. 什么叫做中断嵌套?	261
66. 中断方式 IM0 和 IM2 有什么区别?	267

67. 中断方式 IM1 有什么特点?	273
68. 什么叫做非屏蔽中断 NMI?	274
69. 从中断控制学习中能得到哪些结论?	274
习题	277
第十章 接口电路 PIO	279
70. PIO 的用途是什么?	279
71. PIO 的内部结构怎样? 它怎样与 CPU 和外设相连?	279
72. PIO 的地址信号输入端怎样与 CPU 地址总线相连?	282
73. 怎样编写 PIO 中断字节输出方式的初始化程序?	285
74. 你能编制一个简单的 PIO 中断字节输出方式程序?	292
75. 怎样编制中断字节输入方式的初始化程序?	295
76. 什么叫做字节输入/输出工作方式?	301
77. 怎样编写 PIO 位输入/输出方式的初始化程序?	301
78. 你能为位输入/输出方式控制系统编程?	310
79. 从 PIO 的学习中, 应得到哪些结论?	313
习题与实验	315
第十一章 接口电路 CTC	317
80. CTC 的用途是什么?	317
81. CTC 怎样与 CPU 和外设相连?	318
82. CTC 的地址信号输入端怎样与 CPU 的地址总线相连?	320
83. 怎样编写定时器初始化程序?	322
84. 你能为这些定时器实例编程?	331
85. 怎样编写 CTC 计数器的初始化程序?	338
习题与实验	343
第十二章 微机在控制系统中的应用	345
86. 为什么微机能得到广泛的应用?	345
87. 怎样用微机进行数据采集?	346
88. 微机怎样输出模拟信号?	355
89. 怎样用微机控制三相桥式可控硅整流电路?	363
90. 怎样组成一个价廉物美的微机实时控制系统?	372

习题与实验	380
附录	383
一 Z80微机指令表	383
二 指令对标志位的影响	401
三 ASCII码	404
四 几种八位微机芯片的性能比较	405

第一章 微机的基本概念

内容提要：本章主要讨论微机的基本硬件结构，以及如何组成一个最基本的微型计算机。

1. 在应用微机前，为什么要掌握二进制数和十六进制数？

人们讲话、写作都要使用文字和符号。在我国使用的是汉字，在西方使用的是A、B、C等。这就是说，在两个对象（人和人、人和机器等）之间交换信息时，应有一个共同的语言。我们使用微机，总得找到与微机交换信息的共同语言罢。

在数字电路学习中已知道，一切数字电路都用二进制数字符号0和1，微机也不例外。微机的语言是用二进制符号组合来表示的，微机中的数也是用的二进制数。但是，人类的祖先在公元400年左右就发明了十进制数，大约在公元800年阿拉伯人首先使用它，故称“阿拉伯数”。阿拉伯数一直使用到今天，并仍将使用下去。阿拉伯数与微机使用的二进制数并不完全一样，两者不能直接沟通。

为了使用微机，人们必须首先掌握数制（二进制数、十进制数和十六进制数）的基本规律，以及它们之间的转换方法，这是人机交换信息的基础。

（1）十进制数

十进制数无须多说，因为人们一直在使用它。如323.26，不说也知它是多大的一个数。但是十进制所包含的规律并不是人人都清楚的：

（a）十进制数有十个数字符号，即0、1、2、3、4、

8710626

5、6、7、8、9，任何一个数都是由这十个符号组成。

(b) 十进制计数法是“逢十进一”，如 $9 + 6 = 15$ 就是按逢十进一计算的。

(c) 数字符号在不同位时，它代表的大小不一样。如上面所述的

$323.26 = 3 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 + 2 \times 10^{-1} + 6 \times 10^{-2}$ 也就是说，在小数点前面，各位数的幂次是 10^0 、 10^1 、 10^2 等。而在小数点后面，各位数的幂次是 10^{-1} 、 10^{-2} 、 10^{-3} 等，人们把这里的 10 称为十进制数的“基数”。

为了区分十进制数与将要使用的二进制数和十六进制数，人们常用十进制数英文 Decimal 第一个字母 D 附在数字后面，如 101 D 表示十进制数一百零一。在有些参考书上，为了简化，事先约定，凡数字后无任何字母者均为十进制数。本书也采用这种约定。

(2) 二进制数

自从有了数字电路后，二进制数才获得广泛的应用。如二进制数 101，它不是一百零一，而是二进制数 101（读作 Yao ling Yao）。

对照十进制数，不难找出二进制数的规律：

(a) 二进制数只有两个数字符号 0 和 1，任何一个二进制数都是由这两个数字符号组成。

(b) 二进制计数法是“逢二进一”，如 $11 + 11 = 110$ ，这里应用了逢二进一的概念。

(c) 数字符号在不同位时，它的大小不一样。如 $1011.01 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2}$ 。也就是说，在小数点前面，各位数的幂次递升： 2^0 、 2^1 、 2^2 、 2^3 等。在小数点后面，各位数的幂次递减： 2^{-1} 、 2^{-2} 等。不

言而喻，二进制数的基数是 2。

二进制数英文是 Binary，所以二进制数后都附有 B，以示区别其它数制。如 101B 这表示二进制数 101，而不会误认为十进制数。

(3) 十六进制数

从理论上说，有了十进制数和二进制数已能应付微机的应用。人们用十进制计数，机器用二进制计数，如果再找出它们之间的变换规律，应该说，数制问题就解决了。

为什么还要引入十六进制数呢？这是因为人们读写二进制数实在太费劲了，如 101101010111B，这个数读起来既吃力，写起来也极易出错。从实际工作经验出发，人们又拟定出十六进制数的方案。

十六进制数有什么规律呢？

(a) 十六进制数有十六个数字符号，它们是：

0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F。其中 A 为 10，B 为 11，C 为 12，D 为 13，E 为 14，F 为 15。任何一个十六进制数都是由这十六个数字符号组成。初次应用 A、B、C、D、E、F 来表示数常感不习惯，但是，用多了用久了，一切成自然。

(b) 十六进制计数法是“逢十六进一”。如 $7 + 8 = F$ 无进位； $7 + B = 12$ 有进位。

(c) 数字符号在不同位时，它的大小不一样。如 $3AB.1C = 3 \times 16^2 + A \times 16^1 + B \times 16^0 + 1 \times 16^{-1} + C \times 16^{-2}$ 在小数点前面，各位数的幂递升， 16^0 、 16^1 、 16^2 等。在小数点后面，各位数的幂递减， 16^{-1} 、 16^{-2} 等。不言而喻，十六进制数的基数是 16。

十六进制数的英文是 Hexadecimal，所以十六进制数

后面都附加H，如2000H，这是指十六进制数。如果要化到十进制数，2000H为8192。

除了十进制数、二进制数和十六进制数以外，在日常生活中，人们还使用其它计数形式。如24小时为一天，这就是24进制；十二个月为一年，这就是12进制；我国过去使用的老秤，十六两为一斤，这就是16进制。不同的数制仅仅表现为计数的基数不同。

2. 三种数制之间怎样转换？

我们研究上述三种数制之间的转换规律，目的在于应用微机时能将十进制数转换成二进制数(或十六进制数)输入；微机运算的结果(当然是二进制数)又可转换成十进制数输出。至于其它用途在将来编程时再谈。

(1) 二进制数与十进制数相互转换

(a) 二进制数转换成十进制数 $B \Rightarrow D$

设有一个二进制数101101.11B，它相应的十进制数是多少呢？按照二进制数的幂和基数规则

$$101101.11B = 1 \times 2^5 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} = 45.75$$

由此可见，二进制数转换成十进制数，只要将二进制数中为1的各位按幂相加即可。

(b) 十进制数转换成二进制数 $D \Rightarrow B$

设有一个十进制整数25，它相应的二进制数是多少？其实，对这个较小的数，心算也能求出，

$$25 = 11001B$$

不妨核算一下，

$$11001B = 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^0 = 25$$

上述转换结果无疑是正确的。其实，二转十也有规律可循，

$$25 \div 2 = 12 \quad \text{余 } 1 \text{ (最低位 LSB)}$$

$$12 \div 2 = 6 \quad \text{余 } 0$$

$$6 \div 2 = 3 \quad \text{余 } 0$$

$$3 \div 2 = 1 \quad \text{余 } 1$$

$$1 \div 2 = 0 \quad \text{余 } 1 \text{ (最高位 MSB)}$$

把十进制数依次除 2，并依次记下每次所得的余数（余数总是 1 或 0），不断除下去直到商为 0 为止，全部余数即为相应的二进制数，低位在上，高位在下。

以上是十进制整数化为二进制整数的规律。对于十进制小数，规律有些不同。如 0.3125 等于多少二进制数？可这样计算：

	大于 1 的溢出数
$0.3125 \times 2 = 0.625$	0 (MSB)
$0.625 \times 2 = 0.25$	1
$0.25 \times 2 = 0.5$	0
$0.5 \times 2 = 0.0$	1 (LSB)

$0.3125 = 0.0101\text{B}$ ，这个结果无疑是正确的。由此可见，将十进制小数依次乘 2 得到溢出数，如此乘下去，直到小数部分为零为止，所有溢出数即为相应的二进制小数。

对于一个具有整数和小数两部分组成的十进制数，只需将整数部分和小数部分分别转换，最后用小数点将它们连起来就得到二进制数了。

(2) 十进制数与十六进制数相互转换

(a) 十六进制数转换成十进制数 $H \Rightarrow D$

设有十六进制数 $3A.BH$ ，根据十六进制数规律，其相应的十进制数为：

$$3A.BH = 3 \times 16^1 + A \times 16^0 + B \times 16^{-1} = 58.6875$$

因此，十六进制数转换成十进制数只要将十六进制数各位与相应的幂相乘，最后相加即得十进制数。

(b) 十进制数转换成十六进制数 $D \Rightarrow H$

设有十进制整数 32，其相应的十六进制数为 20H，如果核算一下：

$$20H = 2 \times 16^1 + 0 \times 16^0 = 32$$

结果无疑是正确的。对照十进制整数转换成二进制数的规律，不难得到十进制整数转换成十六进制数的规律：

$$32 \div 16 = 2 \quad \text{余 } 0 \quad (\text{LSB})$$

$$2 \div 16 = 0 \quad \text{余 } 2 \quad (\text{MSB})$$

因此，将十进制数依次除以 16，记下每次所得的余数(余数在 0 ~ F 之间)，不断除下去，直到商为 0 为止，全部余数即为相应的十六进制数，低位在上，高位在下。

如果把十进制小数转换成十六进制小数，只要将十进制小数连续乘以 16，记下溢出部分，直到小数部分为 0 为止。例如求 0.136 相应的十六进制小数：

大于 1 的溢出数

$$0.136 \times 16 = 0.176 \quad 2 \quad (\text{MSB})$$

$$0.176 \times 16 = 0.816 \quad 2$$

$$0.816 \times 16 = 0.056 \quad D$$

$$0.056 \times 16 = 0.896 \quad 0$$

$$0.896 \times 16 = 0.336 \quad E \quad (\text{LSB})$$

$$\dots\dots\dots \vdots$$

故 $0.136 \approx 0.22D0E \quad H$

不言而喻，既有整数又有小数的十进制数，把整数和小数部分分别转换，最后用小数点将它们连起来，就得到十六进制数了。

(3) 二进制数与十六进制数相互转换

二进制数与十六进制数相互转换比较简单，这是因为四位二进制数正好等于一位十六进制数，而且它们的进位规律也相同，如表 1-1 所示。因此，只要将二进制数从低位到高位四位一撇，每四位二进制数换成一位十六进制数，即完成

表1-1 二进制数与十六进制数对应关系

四位二进制数	一位十六进制数
0 0 0 0	0
0 0 0 1	1
0 0 1 0	2
0 0 1 1	3
0 1 0 0	4
0 1 0 1	5
0 1 1 0	6
0 1 1 1	7
1 0 0 0	8
1 0 0 1	9
1 0 1 0	A
1 0 1 1	B
1 1 0 0	C
1 1 0 1	D
1 1 1 0	E
1 1 1 1	F

了二进制数与十六进制数之间的转换。如