高等学校教学参考书

汇编语言程序习题及选解

郭福顺 主编

44
/1

高等教育出版社

高等学校教学参考书

汇编语言程序习题及选解

郭福顺　主编

高等教育出版社

内 容 提 要

本书是为配合计算机有关专业开设"汇编语言程序设计"课程而编写的一本教学参考书．全书共汇集了习题 440 个，一些典型习题给出了解法，以便使学生能更好地掌握程序设计的基本方法和技巧．大部分解法是用 PDP-11 和 DJS-130 两种汇编语言编写的．全书的编排结构基本上对应于"汇编语言程序设计"课程的讲授过程．

本书可供计算机专业的学生、教师以及从事计算机工作的有关人员参考．

责任编辑　纯涌

JS51/25

高等学校教学参考书

汇编语言程序习题及选解

郭福顺　主编

*

高等教育出版社出版

新华书店北京发行所发行

北京印刷三厂印装

*

开本 787×1092 1/16　印张 19.5　字数 440,000

1984 年 1 月第 1 版　1984 年 10 月第 1 次印刷

印数 00,001－30,500

书号 13010·0970　　定价 3.05 元

前　言

本书是根据1980年4月在南京召开的计算机软件教材座谈会的精神，为配合程序设计课而编写的《汇编语言程序习题及选解》。根据我们多年讲授程序设计的经验，考虑到汇编语言程序设计的特点，选编了程序设计的课内习题，课外习题和复习思考题共440个，其中部分习题给了解法。全部习题经过分类编排，分成数制及其转换，数的原码、补码、反码表示，指令练习，简单程序设计，循环程序设计，子程序，输入输出与中断，线性表，浮点解释与浮点运算，递归与再入，解释与模拟，面向系统软件的习题，面向应用的习题，人-机竞赛等十四个部分。为了提高灵活运用基础知识的能力，除选编相当数量的基本练习题外，还选编了相当多的综合练习题。有些题是比较难的，这是为那些想进一步钻研的读者准备的。考虑到程序设计课大都安排在大学低年级，因此，所选习题不涉及"数值分析"方面的知识。

我们选择本书的部分习题，用PDP-11（对应于国产的DJS-180系列）和DJS-130两种汇编语言给出了解法，供读者参考。解法一般由解法分析，程序框图和汇编语言程序三部分组成；解法力求简明易懂，便于自学。一个题目往往有几种解法，本书一般只提供一种解法作为参考。

本书可作为高等院校计算机有关专业汇编语言程序设计课程的教学参考书，对于想了解程序设计技巧的科技工作者或初学程序设计的同志，也可以从本书中得到有益的帮助。

参加本书编写和程序调试工作的有郭福顺（主编），高永慧，李莲治，陈骏林。此外吉朝艳，杨德华，侯莉同志参加了穿孔、汇编和资料整理工作，付出了辛勤的劳动。

中山大学姚卿达副教授仔细审阅了本书初稿，提出了许多宝贵意见。在审稿会上，清华大学，华中工学院，西北电讯工程学院，中国科技大学，华侨大学的代表也提出了许多修改的建议，在此一并表示衷心的感谢！

由于水平所限，况且程序的调试可能不周，书中难免会有缺点和错误，望批评指正。

编　者

1983年5月

目　录

一　数制及其转换……1
二　数的原码、补码、反码表示……4
三　指令练习……7
1. PDP-11 指令练习……7
2. DJS-130指令练习……12
四　简单程序设计……22
五　循环程序设计……32
1. 单重循环……32
2. 多重循环……71
六　子程序……109
七　输入输出与中断……128
八　线性表……147
九　浮点解释与浮点运算……173
十　递归与再入……203
十一　解释与模拟……216
十二　面向系统软件的习题……233
十三　面向应用的习题……253
十四　人-机竞赛……271
附录 1　关于本书所用表示法的若干说明
附录 2　PDP-11指令表
附录 3　DJS-130 指令表
附录 4　第一、二、九章部分习题答案

此为试读，需要完整PDF请访问：www.ertongbook.com

一　数制及其转换

提要：数制及数制转换是在初学计算机原理和汇编语言程序设计时必须认真对待的第一个课题．初学者往往习惯于日常的十进位计数制，而对数字计算机中通常采用的二进位计数制却不太适应．要改变这种状况，唯一的办法是多做题，多接触二进制，八进制，十六进制等计数制，逐渐地就会从不习惯到习惯了．本章的习题就是为熟练掌握各种数制间的转换而选配的．值得注意的是，计算机中的数是二进制表示的．八进制，十进制，或十六进制只是为了人-机之间交换信息简便而引入的．

还应指出，不同的进位制，它们的基数是不同的．1.24 题之后的几个思考题将进一步帮助读者建立进位制的概念．

1.1　将下列二进制整数化成十进制数：

110111，111000，01010101，10101010，

1001000110000001，0011010010000010．

1.2　将下列二进制小数化为十进制数：

0.1010，0.11，0.001，0.11001．

1.3　将下列二进制数化为十进制数：

101010.011，1111110.011001．

1.4　将下列八进制整数化为十进制数：

45，144，1752，22．

1.5　将下列八进制小数化为十进制数：

0.76，0.013，0.0035．

1.6　将下列八进制数化为十进制数：

123.321，20.147，10.045．

1.7　把下列十六进制整数转换成十进制整数：

0064，FF9C，00A0，FF60，0200，2710，

0050，A000，F830，5000．

1.8　将下列十进制整数化为二进制数：

25，69，102，135，284．

1.9　将下列十进制小数化为二进制数(二进制小数点后取十位，十位后舍去)：

0.8750，0.593，0.031，0.370．

1.10　将下列十进制数化为二进制数(二进制小数点后取十位，十位后舍去)：

276.53125，99.4375，53.5625．

1.11　将下列十进制整数化为八进制数：

32, 125, 286, 1100.

1.12 将下列十进制小数化为八进制数:

0.95, 0.001, 0.25, 0.5, 0.625.

1.13 将下列十进制数化为八进制数:

100.01, 64.5, 379.46, 17.21.

1.14 把下列十进制整数转换成十六进制整数:

1100, −100, 160, −160, 5120, 10000,

40000, 80, −24576, 20480.

1.15 试列出 2^n-1 的十进制数和八进制数对照表, 其中 n=0, 1, 2,…, 15.

1.16 把下列十进制数转换为四进制数:

1100, 160, 5120, 10000, 20480.

1.17 试将原码表示的二进制数

10010100. 11001

分别转换成四进制、八进制、十六进制数.

1.18 请将十进制数

180.5625

分别转换成二进制、四进制、八进制、十六进制数.

1.19 把下列各数转换成十进制数:

$(101.10011)_2$, $(123.123)_4$, $(145.353)_6$,

$(22.2)_8$, $(92.11)_{12}$, $(AD.4)_{16}$.

1.20 完成下列各式的二进制运算:

0.11+0.001, 101.011−11.101, 1101×101,

0.1101×0.1011, 101110111÷1101, 0.1011÷11.001.

1.21 试比较下列各数对中的两个数, 指出哪个大? 哪个小?

$(1000)_{10}$ 和 $(1000)_8$, $(0.11)_{10}$ 和 $(0.11)_8$,

$(4096)_{10}$ 和 $(7777)_8$, $(1000)_{10}$ 和 $(809)_{16}$.

1.22 写出 PDP-11 和 DJS-130 计算机全字长正整数的最大值及最小值.

1.23 将 1.8 中各数用二-十进制表示出来.

1.24 如何把一个十进制整数化成 r 进制数(r 为不等于 1 的自然数).

1.25 试写出基数为 3 的头 17 个整数和基数为 9 的头 17 个整数, 并说明基数为 3 和基数为 9 的数之间是什么关系.

1.26 试回答下列问题:

(1) 以 2 为基底的数制有多少不同的数字?

(2) 在以 6 为基底的数制中, 最大的数字是什么?

(3) 一种数制只有零到七这八个数字, 这种数制的基底是什么?

(4) $a_3a_2a_1a_0$为一个十六进制数，与它等值的十进制数是多少？

(5) 和 $357AB_{(16)}$ 等值的二进制数有几位(前面不带 0)？

1.27 指出下列各种说法是对的还是错的：

(1) 任何一个八进制数的位数是等值的十六进制数的一倍；

(2) 十六进制数可以看作是八进制数和二进制数的缩写表示方式；

(3) 十进制数可以不要任何计算工作就转换成十六进制数。

1.28 有一千支铅笔，要求分装在十个盒子里。试问应怎样分装才能适合下述要求：当有人索取时，无论他要领多少支(不超过 1000 支)，都能付给他若干个整盒。

二　数的原码、补码、反码表示

提要：数在计算机中的表示有原码、反码、补码三种形式．原码形式与我们日常使用的数比较起来，除了数制的不同之外，没有什么差别，很容易被人理解．但只有原码形式是不够的，为了使机器做减法时能够简化，还必须采用补码或反码的形式．因此初学者应通过练习，深入理解补码、反码的概念，掌握原码、补码和反码之间的相互转换．为此，我们选编了 28 个习题，其中包括数在机器内的三种表示形式及其计算(2.1～2.14)，证明题(2.15～2.17)和思考题(2.18～2.28)。

2.1　用六位二进制写出下列各数的原码、补码、反码：

10101，11101，11111，−11111，10000，−10000.

2.2　已知下列有符号数的原码，写出其对应的补码和反码：

$[x]_{原}=01001$,

$[x]_{原}=11101$,

$[x]_{原}=11011$.

2.3　已知 x 的补码，求$[x]_{反}$和$[x]_{原}$：

$[x]_{补}=01001$,

$[x]_{补}=11001$,

$[x]_{补}=10000$.

2.4　下列各数是字长为 8 位时的补码，写出其对应的十进制数：

01111000，11110011，10100011，01010100，

00000011，11011001，11111111.

2.5　写出下列十进制数的补码(模 2^7)：

1，16，−16，−128，127，121，90.

2.6　将下列八进制数写成二进制补码形式(mod 2^{15})：

−126，−307，−400，4315，−77777，−1.

2.7　试写出下列用原码表示的八进制数的反码：

175320，057431，000111，111000，177700，077777.

2.8　假设我们要在字长为 6 位的计算机中表示−32 到+31 之间的十进制整数．为此采取下述步骤进行转换：

(a) 把要转换的数 i(−32≤i≤+31)加上模 32；

(b) 将 $i+32$ 转换成二进制整数；

(c) 将二进制表示的 $i+2^5$ 的符号位(权为 2^5 的那一位)取非。

试问：按上述步骤转换的结果是什么形式的数？

试按上述步骤将−32 到+31 之间的全部十进制整数转换成所述形式的数并列出对照表。

2.9 把下列四位二进制表示的补码化成八位二进制表示的补码：

1100, 0101

2.10 回答下列问题：

(1) 在四位二进制补码表示中"1000"表示哪个十进制数？

(2) 写出二进制数"0000"的补码.

(3) 在四位二进制反码表示中，"1000"和"0000"各表示哪个十进制数？

(4) 有符号数"1000"和"0000"为四位二进制表示的原码，它们各代表哪个十进制数？

2.11 设字长为 4，写出 -8 到 7 之间的所有整数的补码.

2.12 设字长为 4：

(1) 写出全部无符号整数.

(2) 写出用补码表示的全部有符号数.

2.13 在以 16 为模的补码系统中计算下列各式：

(1) $6-1$; (2) $7-(-2)$; (3) $-3-3$; (4) $3+4$; (5) $-3+4$.

2.14 在以 16 为模的反码系统中计算下列各式：

(1) $8+8$; (2) $6-1$; (3) $7-11$; (4) $7-(-2)$; (5) $-3-3$.

2.15 已知 x 为二进制数，$[x]_{补}=x_0x_1x_2\cdots x_n$，且 $x_n=0$，求证

$$\left[\frac{1}{2}x\right]_{补}=x_0x_0x_1\cdots\cdots x_{n-1}.$$

2.16 我们知道，对于二进制整数，求负数的补码可以由定义 $[x]_{补}=2^n+x$ 出发求得，也可以按下述方法求得：

(1) 将该负数写成原码形式.

(2) 除符号位外每位取反.

(3) 再在最末位加 1.

试证明两种方法的等价性.

2.17 对于二进制小数，负数的补码可定义为 $[x]_{补}=2+x$. 求负数补码时可以由定义出发求得，也可按上题所述方法求得. 试证明二者的等价性.

2.18 负数在机内有几种表示方法？

2.19 为什么当谈到整数的补码时，总要指出其位数？

2.20 设字长为 8 位. 试问，若用补码表示，则最多能表示多少个不同的数？若用原码表示呢？

2.21 已知 $[0101]_{补}=1011$，$[1011]_{补}=0101$，回答下列问题：

(1) 1011 是哪个十进制数的补码？

(2) 0101 是哪个十进制数的补码？

(3) 对用补码表示的数实行取补操作的实质是什么？

2.22 如何判断一个用补码表示的八进制数是正数还是负数？

2.23 在补码系统中，一个负数取补后应为正数，而$(1000)_2=(-8)_{10}$取补后为1000，仍是一个负数，问题出在哪里？

2.24 用四位二进制原码表示有符号数，其表示范围是多少？若改用补码表示，其表示范围又是多少？

2.25 用八位二进制补码能表示的最大正整数是多少？绝对值最大的负整数又是多少？

2.26 用N位二进制补码能表示多大范围内的整数？能表示$-2^{(N-1)}$和$+2^{(N-1)}$吗？

2.27 在补码系统中，做算术运算时，符号位是否与其它位同样对待？在原码系统中呢？

2.28 试回答下列问题：

(1) 在计算机中，二进制数的加、减、乘、除等算术运算可以归结为加、减、移位三种操作。上面这种说法对吗？

(2) 在计算机中，减法是转化为"加负数"的办法来实现的，也就是"加补码"的办法来实现的。上面这种说法对吗？如何利用"加补码"的办法来代替减法？

(3) 如果(1)、(2)是对的，那么在引入补码之后，二进制数的四则运算就可以归结为加法和移位两种操作了。这对计算机的结构有什么影响？

此为试读，需要完整PDF请访问：www.ertongbook.com

三 指令练习

提要： 根据指令的不同特点，我们在这一部分选编了各种类型的练习题共 42 个，其中 PDP-11 指令练习题 16 个，DJS-130 指令练习题 26 个。这些练习题是为初学这两个指令系统的读者准备的。读者应通过这些指令练习巩固和加深对变址，间接地址，移位，进位，转子和堆栈等概念的理解并熟练掌握各种指令的功能和用法。现将各题侧重练习的内容列表如下，供选题时参考。

PDP-11 指令练习

题　号	侧重练习的内容
2.1～2.2	基本指令
2.3～2.7	汇编指令和机器指令之间的转换
2.8	寻址
2.9～2.10	子程序指令和堆栈
2.11～2.13	转移
2.14～2.16	复习思考

DJS-130指令练习

题　号	侧重练习的内容
2.17～2.23	算术逻辑指令及进位位
2.24～2.30	访内指令
2.31～2.32	汇编指令和机器指令之间的转换
2.33～2.42	寻址

1. PDP-11 指令练习

3.1 试指出下述各程序段完成什么功能：

```
(1) ADD   A0   B0   ;
    ADC   B1        ;
    ADD   A1   B1   ;
(2) NEG   A1        ;
    NEG   A0        ;
    SBC   A1        ;
(3) SUB   A0   B0   ;
```

```
        SBC   B1        ;
        SUB   A1   B1   ;
(4)     NEG   A0        ;
        ADC   A1        ;
        NEG   A1        ;
(5)     SUB   A1   B1   ;
        SUB   A0   B0   ;
        SBC   B1        ;
```

3.2 试阅读下列各程序，给出程序注解，并说明它们完成什么功能。

```
(1)  x=已知数          ;
     y=已知数          ;
     MOV   #x, R1      ;
     ASR   R1          ;
     INC   R1          ;
     ADD   #y, R1      ;
     ADD   #y, R1      ;
     MOV   R1, Z       ;
     HALT              ;
Z:   .WORD  0          ;
(2)  N=已知整数        ;
     MOV   #N, R0      ;
     MOV   R0, R1      ;
     ROR   R0          ;
     BCS   A1          ;
     ASR   R1          ;
     NEG   R1          ;
     HALT              ;
A1:  INC   R1          ;
     ASR   R1          ;
     HALT              ;
```

3.3 假定下面这条指令：

```
MOV  100,   200  ;
```

存放于位置150，试写出汇编结果。如果上述指令改为

```
MOV  #100, 200  ;
```

或 MOV #100, @#200 ;

汇编结果应该是什么?

3.4 试将下述汇编语言程序转换成机器代码程序，并说明该程序完成什么功能。

```
        .=1000
START:  CLR     R0
        MOV     #K,         R1
LOOP:   ADD     (R1)+,      R0
        CMP     #K+18.,     R1
        BPL     LOOP
        HALT
K:      .WORD 1, 2, 3, 4, 5, 6, 7, 10, 11, 12
        .END
```

3.5 试将下述机器语言程序转换成汇编语言程序。

地址	指令
01000	005000
01002	012701
	001020
01006	062100
01010	022701
	001042
01014	100374
01016	00000
01020	00001
01022	00002
01024	00003
01026	00004
01030	00005
01032	00006
01034	00007
01036	00010
01040	00011
01042	00012

3.6 试问下述程序执行完时，A的内容是什么?

```
        .=200
START:  MOV     #230,       R1
```

```
        MOV     2(R1),   R2
        MOV     R1   ,   B
        ADD     B    ,   R2
        MOV     R2   ,   A
        HALT
B:      .WORD   0
C:      .WORD   230,   20,   54
A:      .WORD   0
        .END    START
```

3.7 试问：对上题的程序应作哪些改动才能满足下述要求：

(1) 修改后的程序从 400 处开始汇编；

(2) 当修改后的程序执行完时，A中的内容和上题完全相同.

3.8 设下列寄存器和内存单元的初始内容如下：

(R 0)=0；(R 1)=1；(R 2)=200；(R 4)=400；(R 3)=1000；(R 5)=3000；A=200；B=500；(200)=1000；(230)=30000；(500)=3000；(776)=1000；(1000)=230；(1200)=30020；(3000)=123456；(30020)=1000；

试问执行下列单条指令后，相应的寄存器或内存单元发生什么变化？

```
(a) CLR   (R5)+                    ;
(b) ADD   -(R3),          R0       ;
(c) DEC   @(R2)+                   ;
(d) MOV   #10,            R0       ;
(e) NEG   A                        ;
(f) ADD   (R2)+,          R4       ;
(g) ADD   30(R2),         20(R5)   ;
(h) ADD   @1000(R2),      R1       ;
(i) ADD   @#1200,         R5       ;
(j) SWAB  @               B        ;
```

3.9 (1) 假设栈的内容如图 3.1 所示，试说明执行指令

JSR PC, @(SP)+

后，栈中内容发生什么变化？控制转向哪个单元？

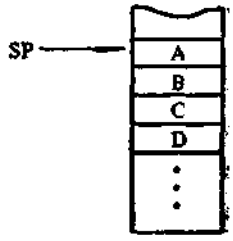

图 3.1

(2) 对下列程序：

```
        MOV     N       ,   R0      ;
        MOV     #1000,      R1      ;
        MOV     #2000,      R2      ;
        MOV     #3000,      R3      ;
```

```
        JSR   R7    ,  ROUTI ;
        HALT                 ;
ROUTI:  MOV   (R1)+ ,  R4    ;
        ADD   -(R2) ,  R4    ;
        MOV   R4    ,  (R3)+ ;
        INC   R0             ;
        BLT   ROUTI          ;
        RTS   R7             ;
N:      -25                  ;
```

(a) 说明每条指令的功能，指出该程序的作用；

(b) 指出每条指令所占内存储单元的数目及访问内存储器的次数。

3.10 对下述主程序段和子程序

```
        .=1000
BEGIN:  MOV   PC    ,  SP    ;
        TST   -(SP)          ;
        MOV   #1    ,  R0    ;
        JSR   R0    ,  SSUB  ;
        .WORD 3              ;
        .=2000
SSUB:   MOV   (R0)+ ,  R1    ;
        DEC   R1             ;
        TST   R1             ;
        BEQ   RETUR          ;
        JSR   R0    ,  SSUB  ;
        .WORD 1              ;
RETUR:  RTS   R0             ;
```

(a) 指明该程序执行前堆栈指针的值和堆栈中的值。

(b) 指明在执行主程序中的指令

```
MOV  #1,  R0
```

之后，堆栈指针的值和堆栈中之值。

(c) 指明在每个堆栈操作(即压入和弹出)之后，堆栈指针的值和堆栈中的值。

(d) 指明在主程序段执行结束时，堆栈指针的值和堆栈中的值。

注：在上述各问中，若答案是不确定的，则用"?"来表示之。

3.11 BR和JMP指令都能通过字节位移使用相对编址吗？这两条指令的异同点何在？

3.12 试比较指令序列

```
CMP  #-77777,  #77777
BMI  LESS;
```

和指令序列

```
CMP  #-77777,  #77777
BLT  LESS
```

的执行结果，它们是否一样？为什么？

3.13 试比较下述三条指令

```
BPL  A
BGT  A
BHI  A
```

的功能，并说明其应用的场合。

3.14 试比较指令 RTI 和 RTS 的功能，它们有什么相似之处？有什么不同？

3.15 PDP-11 指令系统中为什么可以不设专门的输入输出指令？它的输入输出是如何实现的？

3.16 作减法时，产生溢出和最高位产生进位有何不同？各表示什么意思？

2. DJS-130 指令练习

3.17 试说明下述各指令序列完成什么功能：

```
(1) MOVL#  2,  2,  SZC ;
    MOVOR  2,  2,  SKP ;
    MOVZR  2,  2       ;
(2) NEG    2,  2       ;
    COM    2,  2       ;
(3) ADC    1,  1       ;
(4) ADCZL  1,  1       ;
(5) MOV    1,  2       ;
    ADDZL  1,  1       ;
    ADDZL  2,  1       ;
(6) MOVL#  0,  0,  SZC ;
    NEG    0,  0       ;
(7) SUBZL  1,  1       ;
    INCOL  1,  2       ;
    NEG    2,  2       ;
    MOVOL  1,  1       ;
    INC    2,  2,  SZR ;
```


此为试读，需要完整PDF请访问：www.ertongbook.com