

例题习题集

周明德 主编

微型计算机 硬件软件及其应用

清华大学出版社

-44

D/1

微型计算机硬件软件及其应用

例 题 习 题 集

周明德 张淑玲 编著

清华大学出版社

1984

内 容 简 介

本书是与《微型计算机硬件软件及其应用》一书配套的例题和习题集。紧密结合教材中的第一章到第七章，按由简到繁循序渐进的原则编写例题和习题供阅读和练习。例题可作为读者学习和分析、看懂程序用，也提供了一些编程序的方法和技巧；习题可作为练习动手编程序之用。其中大量是经上机检验过的可供实用的应用程序，在实际工作中可直接参考和借鉴。

若阅读教材时对照阅读例题，可加深对教材内容的理解，通过做习题可以巩固理论学习的内容，提高分析问题、解决问题的能力。若有条件把例题和习题上机练习，则效果就更好，也是自学的有效途径。

本书可作为大学参考教材，也可用于微型机培训班或作为自学参考教材。

微型计算机硬件软件及其应用

例 题 习 题 集

周明德 张淑玲 编著

*

清华大学出版社 出版

北京

清华园

北京新华印刷厂 印刷

新华书店北京发行所发行 各地新华书店经售

*

开本：787×1092¹/₁₆ 印张：7¹/₄ 字数：185千字

1984年7月第一版 1984年7月第一次印刷

印数：000001—500000

统一书号：15235·124 定价：0.90元

前 言

《微型计算机硬件软件及其应用》一书出版以来得到了广大读者的关心和爱护。在此，我们对广大读者表示深深的感谢。

学习计算机必须理论与实践紧密结合。除了听课和阅读教材以外，还必须进行大量的练习，有条件的读者一定要多上机。

从一个初学者到能够利用汇编语言编应用程序，必须多看多练。要多看程序，从简单到复杂，开始要能看懂别人的程序，逐步能看出别人的思路和方法，进而能学到一些编程的技巧。另外，也要由浅入深地自己动手编程序，从模仿着手到逐步应用自如，这是要经过反复练习的。有条件者要尽量上机调试。

关于接口电路和接口技术也只有反复练习，特别是动手实践才能学到方法和技巧。

同时，我们也了解到有不少高等学校、不少微机学习班选用《微型计算机硬件软件及其应用》一书作为教材，有些单位还以它作为在职科技干部学习计算机的教材。所以，教学工作也迫切需要有一本例题习题集与之相配合。为了满足广大读者特别是初学者的迫切需要，我们编写了这本例题习题集。

在这本例题习题集中，我们按照由浅入深，由简单到复杂循序渐进的原则，紧密结合教材编写了相当数量的例题和习题供读者阅读和练习。例题可作为读者学习和分析、看懂程序用，也提供了一些编程的方法和技巧；习题可作为读者练习自己动手编程之用。其中有大量经过检验的可供实用的应用程序，供读者在实际工作中参考和借鉴。

习题中有分成 A、B 的题：A 题是例题，我们给出了答案，也用简要的文字说明了解决此题的思路，但没有给出流程图，没有详细的解题步骤，留给读者去思考分析。而 B 题则是同类型的练习题，未给出答案。还有一些题我们没有给答案，也供读者练习用。

张淑玲同志编写了本书第二章中的大部分练习题和做出了答案。其余部分由周明德同志编写。大部分答案曾上机调试过。

由于编者的水平有限，选题仍不够广泛、全面，答案也仅供参考，缺点错误在所难免，敬请读者批评指正。

周明德 张淑玲

1983 年 9 月

目 录

前言

第一部分 习题

第一章 概述.....	1
第二章 Z 80 汇编语言程序设计	5
第三章 Z 80—CPU 的时序	13
第四章 半导体存储器.....	15
第五章 输入和输出.....	19
第六章 中断.....	20
第七章 并行接口电路.....	22

第二部分 例题解答及习题答案

第一章 概述.....	23
第二章 Z 80 汇编语言程序设计	28
第三章 Z 80—CPU 的时序	82
第四章 半导体存储器.....	89
第五章 输入和输出.....	94
第六章 中断	100
第七章 并行接口电路	105

第一部分 习 题

第一章 概 述

题 1.1A

把下列十进制数，转换为二进制、八进制、十六进制和 BCD 数：

- | | |
|-----------|----------|
| ① 135.625 | ② 548.75 |
| ③ 376.125 | ④ 254.25 |

题 1.1B

要求同上，要转换的十进制数如下：

- | | |
|-----------|------------|
| ① 67.375 | ② 936.5 |
| ③ 268.875 | ④ 218.0625 |

题 1.2A

写出下列十进制数的原码、反码和补码表示（用 8 位二进制）：

- | | |
|-------|--------|
| ① +65 | ② +115 |
| ③ -65 | ④ -115 |

题 1.2B

要求同题 1.2A，数据如下：

- | | |
|-------|--------|
| ① +33 | ② +127 |
| ③ -33 | ④ -127 |

题 1.3A

用 16 位二进制，写出下列十进制数的原码、反码和补码表示：

- | | |
|--------|--------|
| ① +120 | ② -120 |
| ③ +230 | ④ -230 |
| ⑤ +999 | ⑥ -999 |

题 1.3B

要求同题 1.3A，数据如下：

- | | |
|--------|--------|
| ① +62 | ② -62 |
| ③ +253 | ④ -253 |
| ⑤ +615 | ⑥ -615 |

题 1.4A

写出下列用补码表示的二进制数的真值：

- | | |
|------------|------------|
| ① 01101110 | ② 01011001 |
| ③ 10001101 | ④ 11111001 |

题 1.4B

要求同题 1.4A，数据如下：

① 01110001

② 00011011

③ 10000101

④ 11111110

题 1.5

运算下列式子（无符号数）

① 01010111

② 11011011

+10011111+00001101

③ 11010111

④ 10100010

+01011110+01110101**题 1.6**

用被乘数左移及部分积右移两种方法，分别做下列无符号数的乘法：

① 1111

② 11011011

×1101×10101101**题 1.7A**

做下列无符号数的除法：

① 111) 100011

② 1010) 11111011

题 1.7B

做下列无符号数的除法：

① 1000) 110000

② 1111) 11000011

题 1.8A

计算下列式子的逻辑“与”、“或”及“异或”的结果：

① 1010 0101

② 1001 1101

0000 11111010 0101**题 1.8B**

要求与题 1.8A 同，式子为：

① 1010 0011

② 1100 0011

1000 00001101 1011**题 1.9A**

在以下几个题中所用的模型机的指令系统如下：

指令种类	助 记 符	机 器 码	功 能
数 据 传 送	LD A, n	3E n	$n \rightarrow A$
	LD H, n	26 n	$n \rightarrow H$
	LD A, H	7C	$H \rightarrow A$
	LD H, A	67	$A \rightarrow H$
	LD A, (n)	3A n	以 n 为地址，把该单元的内容送 A，即 $(n) \rightarrow A$
	LD (n), A	32 n	把 A 的内容送至以 n 为地址的单元， $A \rightarrow (n)$
	LD A, (H)	7E	以 H 的内容为地址，把该单元的内容送 A， $(H) \rightarrow A$
	LD (H), A	77	把 A 的内容送至以 H 的内容为地址的单元， $A \rightarrow (H)$
加 法	ADD A, n	C6 n	$A + n \rightarrow A$
	ADD A, H	84	$A + H \rightarrow A$
	ADD A, (H)	86	A 与以 H 为地址的单元的内容相加， $A + (H) \rightarrow A$

(续)

指令种类	助 记 符	机 器 码	功 能
减 法	SUB n	D6 n	$A - n \rightarrow A$
	SUB H	94	$A - H \rightarrow A$
	SUB (H)	96	$A - (H) \rightarrow A$
逻辑与	AND A	A7	$A \wedge A \rightarrow A$
	AND H	A4	$A \wedge H \rightarrow A$
逻辑或	OR A	B7	$A \vee A \rightarrow A$
	OR H	B4	$A \vee H \rightarrow A$
异 或	XOR A	AF	$A \oplus A \rightarrow A$
	XOR H	AC	$A \oplus H \rightarrow A$
增 量	INC A	3C	$A + 1 \rightarrow A$
	INC H	24	$H + 1 \rightarrow H$
减 量	DEC A	3D	$A - 1 \rightarrow A$
	DEC H	25	$H - 1 \rightarrow H$
无条件转移	JP n	C3 n	$n \rightarrow PC$
	JP Z, n	CA n	$Z = 1, n \rightarrow PC$
	JP NZ, n	C2 n	$Z = 0, n \rightarrow PC$
	JP C, n	DA n	$C_y = 1, n \rightarrow PC$
	JP NC, n	D2 n	$C_y = 0, n \rightarrow PC$
	JP M, n	FA n	$S = 1, n \rightarrow PC$
	JP P, n	F2 n	$S = 0, n \rightarrow PC$
停机指令	HALT	76	停机

在给定的模型机中，若有以下程序，分析在程序运行后累加器 A 中的值为多大。若此程序放在以 10H 为起始地址的存贮区内，画出此程序在内存中的存贮图。

```
LD      A, 20H
ADD     A, 15H
LD      A, 30H
ADD     A, 36H
ADD     A, 1FH
HALT
```

题 1.9B

要求问题 1.9A，程序如下：

```
LD      A, 50H
SUB     30H
LD      A, 10H
ADD     A, 36H
SUB     1FH
```

HALT

题 1.10A

在给定的模型机中，写出用累加的办法实现 15×15 的程序。

题 1.10B

要求同题 1.10A，数据为 20×10 。

题 1.11

在模型机中，用重复相减的办法实现除法的程序如下：

```
LD      A, (M2)          ; M2 为放除数的存贮单元
LD      H, A
XOR     A
LOOP:   LD      (M3), A    ; M3 为放商的存贮单元
LD      A, (M1)          ; M1 为放被除数（或余数）的
                          ; 存贮单元
SUB     H
JP      C, DONE
LD      (M1), A
LD      A, (M3)
INC     A
JP      LOOP
DONE:   HALT
```

若此程序放在以 20H 开始的存贮区，画出它的存贮图。

题 1.12

在模型机中，把二进制数转换为 BCD 码的程序流程图如图 1-1 所示。

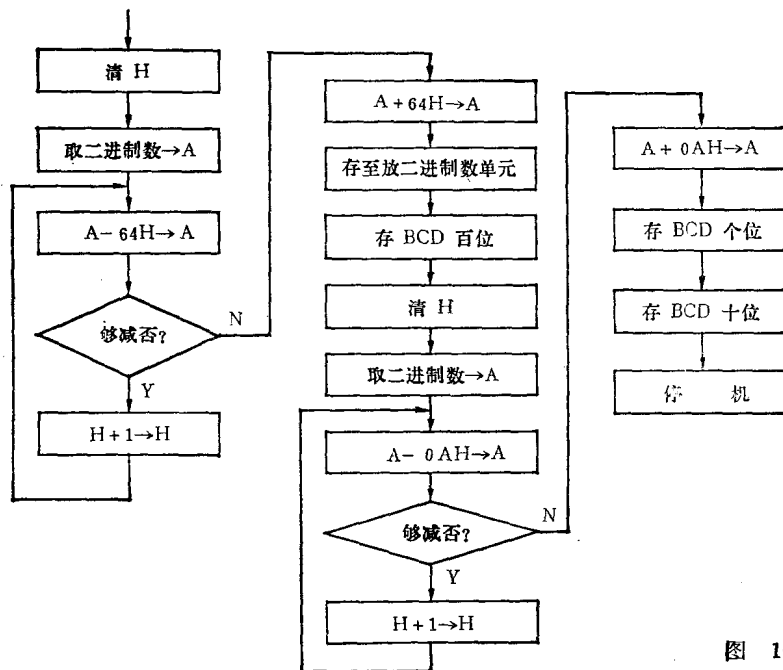


图 1-1

编写出该程序。

第二章 Z80汇编语言程序设计

题 2.1A

在下列程序运行后，给相应的寄存器及存贮单元填入运行的结果：

LD A, 10H			
LD DE, 1000H	B		
LD HL, 2000H	C		
LD (DE), A	D		
EX DE, HL	E		
LD B, (HL)	H		
LD C, 01H	L		
EX DE, HL			
LD (HL), C			
HALT	A		

题 2.1B

要求同题 2.1A，程序如下：

```
LD A, 50H
LD IX, 1000H
LD HL, 2000H
LD (IX), A
LD B, 20H
LD (HL), B
LD C, 01H
LD C, (HL)
LD DE, 3000H
HALT
```

题 2.2

若在自 1000H 单元开始有一个 100 个数的数据块，要把它传送到自 2000H 开始的存贮区中去，用以下三种方法，分别编制程序：

1. 不用数据块传送指令；
2. 用单个传送的数据块传送指令；
3. 用数据块成组传送指令。

题 2.3A

利用变址寄存器，编一个程序，把自 1000H 单元开始的 100 个数传送到自 1070H 开始的存贮区中去。

题 2.3B

要求同 2.3A，源地址为 2050H，目的地址为 2000H，数据块长度为 50。

题 2.4

编一个程序把自 1000H 单元开始的 100 个数, 传送至 1050H 开始的存储区中 (注意: 数据区有重叠)。

题 2.5A

在自 0500H 单元开始, 存有 100 个数。要求把它传送到 1000H 开始的存储区中, 但在传送过程中要检查数的值, 遇到第一个零就停止传送。

题 2.5B

条件同题 2.5A, 但在传送过程中检查数的值, 零不传送, 不是零则传送至目的区。

题 2.6

把在题 2.5A 中指定的数据块中的正数, 传送到自 1000H 开始的存储区。

题 2.7

把在题 2.5A 中指定的数据块中的正数, 传送到自 1000H 开始的存储区; 而把其中的负数, 传送到自 1100H 开始的存储区。且分别统计正数和负数的个数, 分别存入 1200H 和 1201H 单元中。

题 2.8A

自 0500H 单元开始, 有 10 个无符号数, 编一个程序求这 10 个数的和 (用 8 位数运算指令), 把和放到 050A 及 050B 单元中 (和用两个字节表示), 且高位在 050B 单元。

题 2.8B

自 0200H 单元开始, 有 100 个无符号数, 编一个程序求这 100 个数的和 (用 8 位数运算指令), 把和放在 0264H 和 0265H 单元 (和用两字节表示), 且高位在 0265H 单元。

题 2.9

题同 2.8, 只是在累加时用 16 位运算指令编程序。

题 2.10A

若在 0500H 单元中有一个数 x :

1. 利用加法指令把它乘 2, 且送回原存储单元 (假定 $x * 2$ 后仍为一个字节);
2. $x * 4$;
3. $x * 10$ (假定 $x * 10 \leq 255$)。

题 2.10B

题意与要求同题 2.10A, 只是 $x * 2$ 后可能为两个字节。

题 2.11

若在存储器中有两个数 a 和 b (它们所在地址用符号表示, 下同), 编一个程序实现 $a * 10 + b$ ($a * 10$ 以及 “和” 用两字节表示)。

题 2.12

若在存储器中有数 a 、 b 、 c 、 d (它们连续存放), 编一个程序实现

$$((a * 10 + b) * 10 + c) * 10 + d \quad (\text{和} \leq 65535)$$

题 2.13A

在 0100H 单元和 010AH 单元开始, 存放两个各为 10 个字节的 BCD 数 (地址最

低处放的是最低字节), 求它们的和, 且把和放在 0114H 开始的存贮单元中。

题 2.13B

在 0200H 单元和 020AH 单元开始, 存放两个各为 10 个字节的二进制数(地址最低处放的是最低字节), 求它们的和, 且把和放在 0214H 开始的存贮单元中。

题 2.14

在 0200H 单元开始放有数 A (低位在前)

NA=95 43 78 62 31 04 56 28 91 01

在 020AH 单元开始放有数 B

NB=78 96 42 38 15 40 78 21 84 50

求两数之差, 且把差值送入自 0200H 开始的存贮区。

题 2.15A

若在 0500H 单元有一个数 x, 把此数的前四位变 0, 后四位维持不变, 送回同一单元。

题 2.15B

条件同题 2.15A, 要求最高位不变, 后 7 位都为 0。

题 2.16A

若在 0500H 单元有一个数 x, 把此数的前四位变“1”, 后四位维持不变, 送回同一单元。

题 2.16B

把 x 的最低位变“1”, 高 7 位不变, 送至 0600H 单元。

题 2.17A

若在 0500H 单元有一个数 x, 把此数的前四位变反, 后四位维持不变, 送回同一单元。

题 2.17B

把 x 的最高位取反, 后 7 位不变, 送至 0600H 单元。

题 2.18

从 0200H 单元读入一个数, 检查它的符号, 且在 0300H 单元为它建立一个符号标志 (正为 00, 负为 FF)。

题 2.19A

若从 0200H 单元开始有 100 个数, 编一个程序检查这些数, 正数保持不变, 负数都取补后送回。

题 2.19B

把题 2.19A 中的负数取补后送至 0300H 单元开始的存贮区。

题 2.20A

若在 0200H 和 0201H 单元中有一个双字节数, 编一个程序对它们求补。

题 2.20B

在 BC 寄存器对中有有一个双字节数, 对它求补。

题 2.21

若在 0200H~0203H 单元中有一个四字节数, 编一个程序对它求补。

题 2.22A

若在 0200H 和 0201H 单元中有两个正数，编一个程序比较它们的大小，把大的数放在 0201H 单元中。

题 2.22B

条件同上，把较小的数放在 0201H 单元中

题 2.23

条件与要求同题 2.22A，只是是两个无符号数。

题 2.24

条件与要求同题 2.22A，相比较的是两个带符号数。

题 2.25A

若自 0500H 单元开始有 1000 个带符号数，把它们的最小值找出来，放在 1000H 单元中。

题 2.25B

若自 1000H 单元开始有 1000 个无符号数，把它们的最大值找出来，放在 2000H 单元中。

题 2.26

若在 0200H 单元中有一个数 x ，用移位方法实现：

1. $x * 2$

2. $x * 4$ ($x * \leq 255$)

且送回原单元。

题 2.27A

编一个程序使寄存器对 BC 中的数整个左移一位。

题 2.27B

编一个程序使寄存器对 HL 中的数整个右移一位（最高位维持不变）。

题 2.28A

在 0200H 单元中有一个数 x ，利用移位和相加的办法，使 $x * 10$ （假定 $x * 10 \leq 255$ ）后送回原单元。

题 2.28B

条件和要求同题 2.28A，但 $x * 10$ 可大于 255。

题 2.29

在 0200H 和 0201H 单元中，存有一个两字节数（高位在后），编一个程序把它们整个右移一位。

题 2.30

在自 BUFFER 单元开始，放有一个数据块，BUFFER 和 BUFFER+1 单元中放的是数据块的长度，自 BUFFER+2 开始存放的是以 ASCII 码表示的 10 进制数码，把它们转换为 BCD 码，且把两个相邻单元的数码并成一个单元（地址高的放在高四位）。放到自 BUFFER+2 开始的存贮区中。

题 2.31

在自 BUFFER 单元开始，放有一个数据块，BUFFER 和 BUFFER+1 单元中放

的是数据块的长度，自 BUFFER+2 开始存放数据，每一单元放的是两位 BCD 码，把它们分别转换为 ASCII 码，放到自 BLOCK 开始的存贮区中（低四位 BCD 码转换成的 ASCII 码放在地址低的单元），而 BLOCK 和 BLOCK+1 放转换成的 ASCII 码的长度。

题 2.32A

在自 BUFFER 单元开始，放有一个数据块，BUFFER 和 BUFFER+1 单元中放的是数据块的长度，自 BUFFER+2 开始存放的是以 ASCII 码表示的 16 进制数码（即 0—9，A—F），把它们转换为 16 进制数码，放在同一单元中。

题 2.32B

条件同题 2.32A，把转换以后的两个相邻的 16 进制数并在一个存贮单元中。

题 2.33

若在某存贮区中已输入 4 个以 ASCII 码表示的 16 进制数码（高位在前），把它们转换为二进制数放入 HL 寄存器对中。

题 2.34

在自 BUFFER 单元开始的数据块中，前两个单元放的是数据块的长度，自 BUFFER+2 开始存放的是二进制的数。把每一个存贮单元的两位 16 进制数，分别转换为各自的 ASCII 码，放到自 BLOCK 开始的存贮区中（开始两个单元放新的数据块的长度）。

题 2.35

在题 2.30 中，把相邻单元的两个数码看成是两位十进制数（后面的为十位数），把它们转换为相应的二进制数，放到自 BUFFER+2 开始的存贮区中。

题 2.36

在题 2.34 中，把数据块中的每一单元的二进制数转换为相应的 BCD 码（每一字的二进制数，对应三位 BCD 码），再把它们转换为 ASCII 码放到自 BLOCK 开始的存贮区中（开始两个单元放新的数据块的长度）。

题 2.37

若在 DE 寄存器对中，放有四位 BCD 码，把它们转换为相应的二进制数，放在 HL 寄存器对中。

题 2.38A

若在 HL 寄存器对中有一个 16 位无符号数，把它们转换为相应的 BCD 码，放到自 DATA 开始的存贮区中（每一位 BCD 码占一个存贮单元，高位在前）。

题 2.38B

若在 HL 寄存器对中有一个 16 位带符号数，把它们转换为相应的 BCD 码，放在自 DATA 开始的存贮区中（符号占一个单元，每一位 BCD 码占一个存贮单元，高位在前）。

题 2.39

若自 STRING 单元开始存放一个字符串（以字符 \$ 结尾）：

1. 编一个程序统计这个字符串的长度（不包括 \$ 字符）；
2. 把字符串的长度，放在 STRING 单元，把整个字符串往下移两个存贮单元。

题 2.40

若自 STRING 单元开始存放一个字符串（以字符空格引导，以 \$ 结尾）编一个程序统计这个字符串的长度（忽略前导空格和结尾的 \$ 字符）。

题 2.41A

在题 2.40 的字符串中，统计数字字符（'0'~'9'）的个数。

题 2.41B

在题 2.40 的字符串中，统计 16 进位字符（即 '0'~'9', 'A'~'F'）的个数。

题 2.42A

在题 2.40 的字符串中，把十进制数字字符（'0'~'9'）传送至 DATA 开始的存储区中，在 DATA 和 DATA+1 单元存放的是这个数字字符串的长度，自 DATA+2 单元开始存放字符。

题 2.42B

条件和要求同题 2.42A，只是传送的是 16 进制数字字符。

题 2.43

在自 STR1 和 STR2 开始各有一个由 10 个字符组成的字符串，检查这两个字符串是否相等，在 STFLAG 单元中建立一个标志（相等为 00，不等为 FF）。

题 2.44A

编一个程序统计一个八位二进制数中的为“1”的位的个数。

题 2.44B

编一个程序统计一个 16 位二进制数中的为“1”的位的个数。

题 2.45

在自 STR1 开始有一个字符串（前两个字节为字符串长度），对每一个字符配上偶检验位，送回原处。

（以下各题所编的程序必须包含必要的伪指令）

题 2.46

自 NUMBER 单元开始，放有两个多字节的用 BCD 码表示的十进制数，NUMBER 单元放的是字节数，NUMBER+1 开始连续存放两个多字节数（高位在后），编一个程序把这两个多字节数相加，和接着原来的数连续存放。

题 2.47A

编一个程序使放在 DATA 及 DATA+1 单元的两个八位带符号数相乘，乘积放在 DATA+2 及 DATA+3 单元中（高位在后）。

题 2.47B

编一个程序使放在 DATA（被除数）及 DATA+1（除数）的两个八位带符号数相除，商放在 DATA+2 单元，余数放在 DATA+3 单元中。

题 2.48A

编一个程序使放在 DATA 和 DATA+1 的两字节无符号数与在 DATA+2 及 DATA+3 中的无符号数相乘，乘积接着原来的数存放（高位在后）。

题 2.48B

编一个 24 位无符号数相乘的程序。

题 2.49

编一个程序使上题中的两个 16 位无符号数相除，商和余数接着原来的数存放(先放商，高位在后)。

题 2.50

在题 2.48 中，相乘的是两个 16 位的带符号数，计算乘积。

题 2.51

若在自 DATA 开始的 7 个单元中，第一个单元放数的符号（用 ASCII 码表示），接着最多为 5 位用 ASCII 码表示的十进制数码（高位在前），且数值在 ± 32768 范围内，然后是非数字字符（表示数的结束）。编一个程序把它转换为相应的 16 位二进制数（用补码表示），放在 HL 寄存器对中。

题 2.52

把在 HL 寄存器对中的 16 位带符号二进制数，转换为十进制数。且把符号位及各位十进制码转换为相应的 ASCII 码，放到自 DATA 开始的 7 个存储单元中（高位在前），最后填以空格。

题 2.53

在自 STRING 开始有一个字符串（以 '#' 号作为字符串的结束标志），编一个程序，查找此字符串中有没有字符 \$，有多少个 \$（放在 NUMBER 单元中，没有 \$ 则其为 0，否则即为 \$ 的个数）；且把每一个 \$ 字符所存放的地址，放到自 POINTR 开始的连续的存储单元中。

题 2.54A

某一个监控程序中，有 10 个命令，分别以字母 A、B、C、D、E、F、G、H、I、J 表示。这 10 个命令有 10 个处理程序，它们的入口地址形成一个表格 CMDTBL。

3000	CMDTBL	
3000	3500	; A 命令入口
3002	3550	; B 命令入口
3004	3600	; C 命令入口
3006	3640	; D 命令入口
3008	3670	; E 命令入口
300A	36B0	; F 命令入口
300C	3700	; G 命令入口
300E	3730	; H 命令入口
3010	3760	; I 命令入口
3012	37A0	; J 命令入口

若打入的命令字已在累加器 A 中，编一个程序，根据输入的命令字转至相应的处理程序。

题 2.54B

条件同题 2.54A，只是有一个转移指令表，如下所示。

3000	CMDJPT	
3000	JP 3500	; 转至 A 命令处理程序入口
3003	JP 3550	; 转至 B 命令处理程序入口

3006	JP 3600	; 转至 C 命令处理程序入口
3009	JP 3640	; 转至 D 命令处理程序入口
300C	JP 3670	; 转至 E 命令处理程序入口
300F	JP 36B0	; 转至 F 命令处理程序入口
3012	JP 3700	; 转至 G 命令处理程序入口
3015	JP 3730	; 转至 H 命令处理程序入口
3018	JP 3760	; 转至 I 命令处理程序入口
301B	JP 37A0	; 转至 J 命令处理程序入口

要求同题 2.54 A。

题 2.55A

某一个操作系统有 6 个内部命令，把这 6 个内部命令及相应的处理程序的入口地址，组成一个内部命令表 INCMDT，如下所示：

	ORG	2000H
INCMDT:	DB	'ATTRIB'
	DB	00
	DW	2020
	DB	'BYE'
	DB	00
	DW	2050
	DB	'DIR'
	DB	00
	DW	20A0
	DB	'ERA'
	DB	00
	DW	2100
	DB	'REN'
	DB	00
	DW	2140
	DB	'SAVE'
	DB	00
	DW	2180
	DB	'TYPE'
	DB	00
	DW	21E0

若输入的内部命令字符（以 00 字节作为命令字符的结束标志），已在 BUFFER 开始的存储区中。编一个程序根据输入的命令字转至相应的处理程序。

题 2.55B

条件同题 2.55 A，但在 BUFFER 开始的存储区中的命令字，可能是一个内部命令，则应转至相应的处理程序；可能不是内部命令，则在程序中应建立一个标志。