

张秀琼 严化南 杨延双 编著

TP 805 TS 微型机 教学实验系统指导书



地震出版社

TP805 TS微型机教学 实验系统指导书

张秀琼 严化南 杨延双 编著

地震出版社

1988

内 容 提 要

本书是北京工业大学最近推出的 TP805 TS 教学实验系统的指导书。本书共编排了28个实验和练习,包括:TP805微型机的各种操作命令的练习;典型汇编语言程序的调试;存储器扩充;EPROM 编程;各种 I/O 接口芯片与主机的联接及其编程;各种外围设备:显示器、打印机、CRT 等驱动程序的编制和调试;接口的应用实例等。本书中所列举的实验,都经过北工大计算机科学系的教学实践验证过。实验内容系统丰富、有启发性。它不仅适用于教学,也是科研部门开发微型机控制系统的得力工具。

读者对象:适用于计算机专业的本科生和工程技术人员。适当选取内容后也可适用于非计算机专业的本科、专业、中专学生,也可作为各种微型机应用、接口通讯培训班的实验教材和科技人员的培训教材。

JS 458/34 12

TP805 TS微型机教学 实验系统指导书

张秀琼 严化南 杨延双 编著
责任编辑:张晓梅

北京出版社 出版

北京复兴路63号

中国农业机械出版社印刷厂印刷

新华书店北京发行所发行

全国各地新华书店经售

●

787×1092 1/16 10.75印张 270千字

1988年6月第一版 1988年6月第一次印刷

印数00001—15000

ISBN 7-5028-0118-9/T·3

(520) 定价:3.00 元

前 言

北京工业大学在1981年向全国推出TP80 TS教学实验系统以来,受到各大专院校和科研生产单位的欢迎。TP80 TS教学实验系统是由TP801单板计算机控制的教学实验系统,由于受到TP801单板机本身硬件和软件资源的限制,不能方便地使用编辑/汇编、BASIC…等高级语言进行教学实验、产品研制开发和软件调试,只能使用机器语言进行软件编制、软件调试,工作效率甚低。

1983年我校研制、生产出TP805微型计算机,它能运行编辑/汇编和控制BASIC…等高级程序语言,本身带有打印机接口、CRT显示接口和存贮器I/O扩充ROM卡接口电路,它配备有多种应用软件和扩展板。为了提高教学实验、科研开发的工作效率,我们研制出由TP805微型机控制的TP805 TS教学实验系统。

TP805 TS教学实验系统是由实验板、实验器材和实验指导书等组成。该系统充分利用TP805微型机的软件和硬件资料,构成了一套完整的微型机实验、开发装置。

TP805 TS实验指导书,总共编排了28个实验。由浅入深共分三部分:第一部分是由实验一至实验六。这部分实验的目的是帮助用户了解和掌握TP805编辑/汇编语言程序的各种操作命令、各种典型的汇编程序、打印驱动程序、显示器驱动程序等的编制以及TP805 TS板的结构和板上器件的使用,是掌握TP805微型机、熟悉Z80汇编语言程序的基础实验。可以做为《汇编语言程序》课程实验,或者作为TP805用户培训班的教学实验;第二部分是由实验七至实验二十五。这部分是微型机接口实验。从简单的I/O接口、存贮器扩充到多种可编程接口实验。如Z80的多种外围LSI接口芯片PIO、CTC、DMA、SIO…等的基本功能以及应用实验。这部分的内容较为丰富充实,是本教学实验系统的核心实验。不仅可以配合《微型机应用》、《接口与通讯》等课程的教学实验,而且也有很大的实用价值;第三部分是由实验二十六到实验二十八。这是综合性实验。硬件和软件的规模都较前两部分要大得多,可以做为课程设计的教学内容或者作为微型机应用范例。

TP805 TS教学实验系统中的教学实验板,从设计方面做了精心考虑,以使用户更好地开展教学、科研实验工作。与TP80 TS教学实验板比较,板上“面包板”由4块增加到6块,LED和开关由10路增加到17路。改进了教学实验板与主机的联接。只需一根40芯扁电缆一头插入TP805主机总线扩展口,另一头插入教学实验板,就可以将TP805主机上的全部信号线引至实验板上。此外,TP805 TS教学实验板上的接线插孔也做了改进,采用40个单孔带锁紧的天线插孔座。使接线更加方便可靠,维修简便。

我们考虑到TP用户很多都拥有TP80 TS教学实验系统,为了提高这个旧的教学实验系统的技术水平而不至增加太多的投资,充分利用原有的教学实验板,用户只需增加一套TP805微型机代替原来的TP801,再购置一些实验用的LSI芯片,如Z80A CTC、PIO、SIO、DMA、HM6264 (SRAM, 8kB/片)、EPROM2764…等。参照TP805 TS教学实验系统,将TP805微型机上的总线扩展口EXP/40PIN插座,通过一根40芯扁平电缆和TP80 TS实验板上的插座联接,再将TP80 TS板上的插座引脚,按照TP805 EXP/40PIN插座上的信号线重新定义。经过这

样简单的改造后，就可以构成一个以TP805微型机为主机的教学实验系统了。实验用本教学实验系统的实验指导书。

实验指导书中所列举的实验，都经过本校计算机科学系的教学实践验证过。实验参考程序从教学效果上考虑，把它编在指导书的“附录B实验参考程序”。

参加本教学实验系统研制、教材编写、程序调试等工作的有：张秀琼、严化南、杨延双、脱小明、任强等。袁峰、刘大军、刘小敬等曾以此为毕业设计的课题，进行了大量的工作。实验指导书由吴定荣审定。计算机科学系计算机应用教研室的教师们曾大力支持本系统的工作，在此一并致谢。

张秀琼

1987年12月

目 录

实验一 熟悉教学实验板结构实验	(1)
实验1.1 发光二极管(LED)显示实验	(1)
实验1.2 用与非门组成异或门	(3)
实验1.3 组成一个时钟发生器	(4)
实验1.4 消除开关的抖动	(5)
实验二 TP805微型机的操作	(6)
练习2.1 程序的导入和连续执行	(6)
练习2.2 单步执行程序	(7)
练习2.3 设置断点	(8)
实验三 汇编程序的调试	(9)
练习3.1 数据块的传送和检索	(9)
练习3.2 8位二进制带符号数相乘	(10)
练习3.3 打印源程序及结果数据	(11)
实验四 子程序	(13)
练习4.1 子程序、调用程序的编写、调试及参数传送	(13)
练习4.2 TP805系统子程序的调用	(14)
练习4.3 子程序的递归调用	(15)
实验五 图形(汉字)打印	(18)
实验六 CRT显示实验	(23)
实验6.1 CRT字符显示	(28)
实验6.2 CRT图形显示	(31)
实验6.3 CRT半图形(汉字)显示	(34)
实验七 LED七段显示器接口实验	(38)
实验八 存储器扩充实验	(41)
实验九 Z80A PIO接口实验(1)	(43)
实验十 Z80A PiO接口实验(2)	(46)
实验十一 Z80A CTC接口实验(1)	(51)
实验十二 Z80A CTC接口实验(2)	(53)
实验十三 数-模(D/A)转换实验(1)	(55)
实验十四 数-模(D/A)转换实验(2)	(57)
实验十五 模-数(A/D)转换实验(1)	(59)
实验十六 模-数(A/D)转换实验(2)	(61)
实验十七 模-数(A/D)转换实验(3)	(63)
实验十八 DMAC性能(存储器↔存储器)实验(1)	(66)

实验十九	DMAC性能(存储器 $\leftrightarrow$ I/O)实验(2).....	(68)
实验二十	DMAC性能(I/O $\leftrightarrow$ I/O)实验(3)	(69)
实验二十一	DMAC性能实验(4)	(70)
实验二十二	利用Z80A SIO实现两机的异步方式无规约通讯.....	(71)
实验二十三	利用Z80A SIO实现两机的异步方式XON/XOFF规约通讯	(75)
实验二十四	利用Z80A SIO实现两机的异步方式CB信号规约通讯	(78)
实验二十五	利用Z80A SIO的SDLC方式实现多机通讯.....	(82)
实验二十六	交通信号灯实时控制系统.....	(84)
实验二十七	微型机工业控制系统	(88)
实验二十八	EPROM编程器实验.....	(92)
附录A	实验所使用的集成电路芯片引脚图和编程摘要.....	(105)
附录B	实验参考程序清单	(131)
附录C	TP80 TS引脚重新定义表	(163)

实验一 熟悉教学实验板结构实验

实验目的

1. 了解教学实验板的结构;
2. 掌握在接口实验中常遇到的一些最基本的逻辑单元的用法。例如, 如何用发光二极管表示不同的逻辑状态; 消除开关(或按键)抖动电路的接法; 如何产生时钟信号等。

TP805 TS实验板说明

图1.1为TP805 TS实验板的结构图。TP805 TS的实验板母板由印刷电路板制成。母板上标有 (A_0 — A_{15}) 16根地址线, (D_0 — D_7) 8根数据线, 12根控制线 ($\overline{INT}$, $\overline{WAIT}$, $\overline{BUSERQ}$, $\overline{BUSERQ}$, $\overline{M}_1$, $\overline{WR}$, $\overline{MREQ}$, $\overline{RD}$, $\overline{NMI}$, $\overline{IORQ}$, ϕ , $\overline{RESET}$) 以及+5V电源线和地线。这些信号线均与实验板上方的40引脚的插座相接, TP805通过40引脚插座向实验板提供这些信号。插座上各插针的定义和TP805机壳后面右侧的40引脚扩展口插座“EXP”一致。因此通过40芯插头和40根扁平电缆导线, 可以把TP805为扩展用的40根信号线引到实验板上。板的上方有40个具有锁紧能力的单芯导线插孔, 在插孔旁边标有信号线的名称。因此实验者从插孔上可以获得实验所需要的来自TP805的各种信号。右边两排三个插孔是电源, 三个插孔是地。其中一个标有GND的和TP805主机的地联接。板上装有六块元器件插座板(俗称“面包板”)。此教学实验板除作为教学实验外, 还可以作为科研部门开发各种新线路的调试板。

实验板上设有17个“乒乓”开关, 供实验者作数字量输入用。从插孔上可以获得开关量输入信号。

板上设有17个LED发光二极管, 供实验者作显示数字量输出。

此外, 实验板上还装有两个按钮开关, 为实验者提供触发信号。还有100k Ω 和22k Ω 可调式电位器各一个, 作为电压连续无级可调输出。

实验板的电源可来自TP805, 但注意板上总负载不能超过1A。如果实验线路的规模比较大, 应单独向实验板供电。故实验板上设有六个接线柱, 可接 $\pm 5V$ 和 $\pm 15V$ 电源。用户可以很方便地将电源引至板上。因为接线柱并不与面包板相接, 因此, 可以根据需要选用任何电压的电源。在使用 $\pm 15V$ 电源的情况下, 应十分注意+5V电源线千万不要与 $\pm 15V$ 相接触, 否则将毁坏器件。

实验1.1 发光二极管(LED)显示实验

实验目的

学会如何驱动一个发光二极管(简称LED)显示。

实验器材

TP805 TS或TP80 TS实验板一块, 74LS04六反相器1片, 1k Ω 电阻一个。

实验内容和步骤

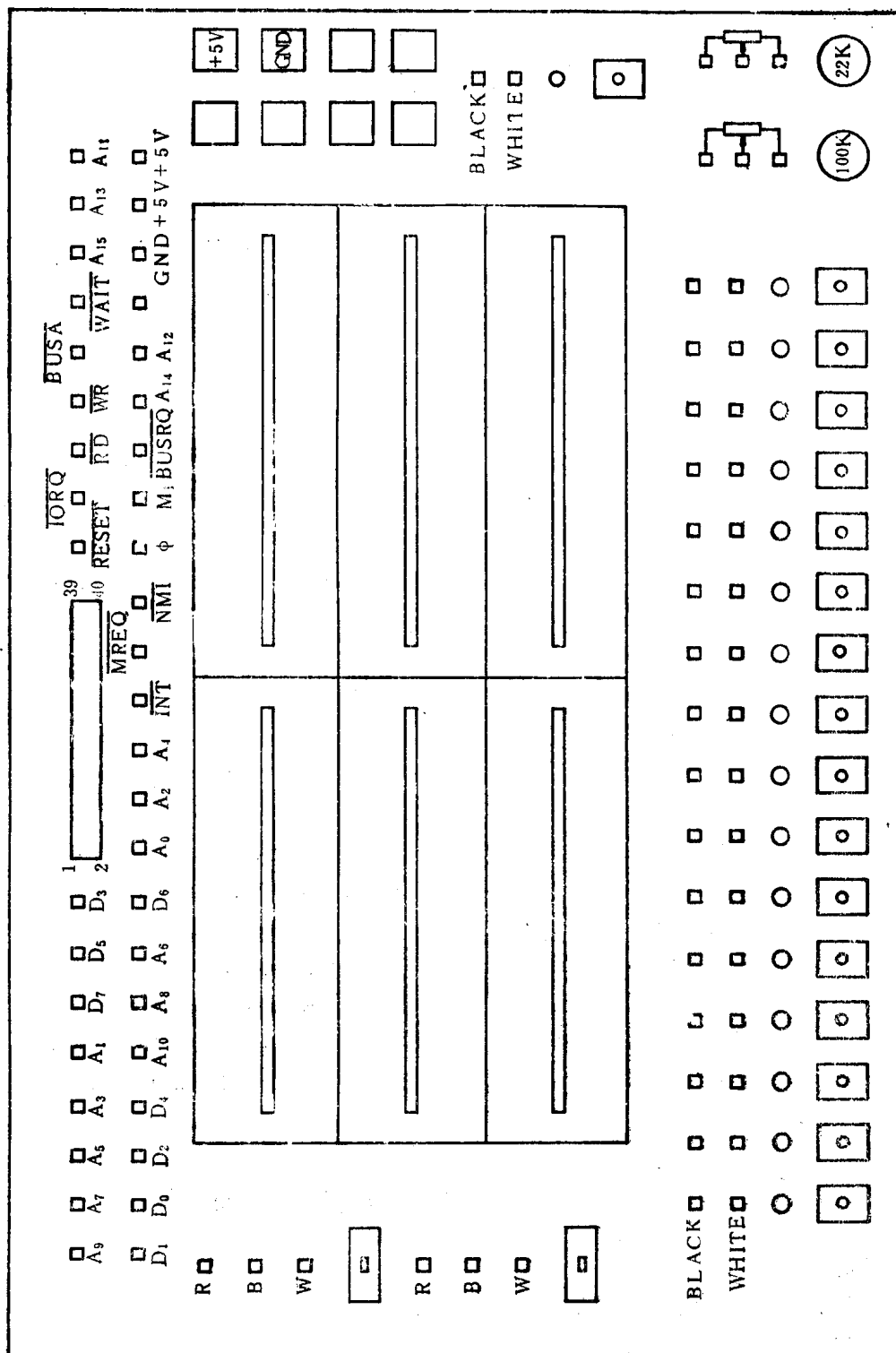


图1.1 TP805 TS实验板结构图

1. 在实验板上已经装了17个LED。观察一下每个LED，接法如图1.2所示。
2. 检查一下R的阻值，是否在1k左右。这里为什么要有一个电阻？
3. 给实验板加上5V电源。用一导线将LED阴极（已引到实验板上的插孔）接地，观察此时LED是否发光？

4. 如果想检测某点的逻辑电平，可以通过被测点连接一个反相器去驱动LED，电路如图1.3所示。

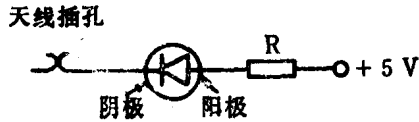


图1.2 LED电路连接图



图1.3 LED驱动实验

LED亮表示被测点逻辑电平为“1”，LED暗表示被测点逻辑电平为“0”。请解释为什么？

5. 用下面的实验可以验证步骤4的正确性。将反相器输入端直接接地（如图1.4），LED应不亮；将反相器输入端通过1k限流电阻接+5V，LED应亮。

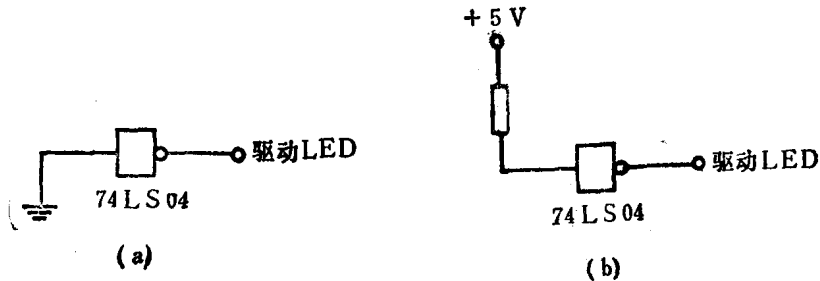


图1.4 LED驱动电路

实验1.2 用与非门组成异或门

实验目的

学会只用与非门组成异或（XOR）门。并用实验的方法用开关作为输入，用图1.3的LED电路作为输出，来验证这种异或关系。

实验器材

TP805 TS或TP80 TS实验板一块，74LS00四与非门1片，1kΩ电阻2个。

实验内容和步骤

1. 请你用与非门组成一个异或电路。

假设异或门的输入为X、Y，输出为Z，异或关系可以用表达式 $Z = \overline{X}Y + X\overline{Y}$ 表达。

2. 用与非门按图1.5连线，可以得到输入X、Y与输出Z之间的异或关系：

$$Z = \overline{\overline{X}Y \cdot \overline{X}\overline{Y} \cdot \overline{X}Y \cdot X\overline{Y}} = \overline{X}Y + X\overline{Y}$$

3. 可以用实验板上的两个乒乓开关作为输入X和Y。实验板上共有17个这样的开关，它们接线如图1.6。

如果把S打在上边，则插孔为低电平；如果把S打在下边，则插孔为高电平。用一根导

线从插孔引到X或Y,把S打上或打下,就会获得“0”、“1”信号。

4. 用开关 S_1 作为X输入, S_2 作为Y输入,再把异或电路输出Z接到LED驱动器(74LS04)的输入。如果扳动 S_1 , S_2 能满足表1.1所示的逻辑关系,那么说明上述电路确实是一个异或电路。

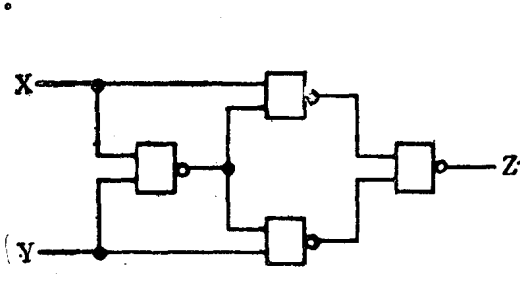


图1.5 用与非门组成异或门线路

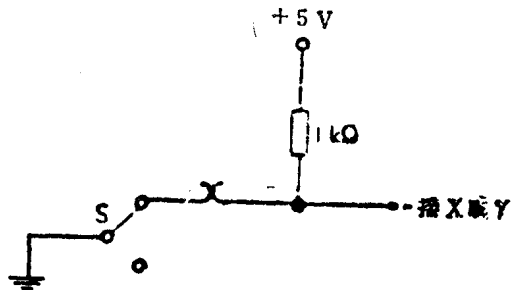


图1.6 板上乒乓开关接线图

表 1-1

S_1 (X)	S_2 (Y)	LED	S_1 (X)	S_2 (Y)	LED
0	0	不亮	1	0	亮
0	1	亮	1	1	不亮

思考题参考答案

$Z = \overline{X}Y + X\overline{Y}$ 用逻辑代数简化: $\overline{X}$ 代用因子为 $\overline{X}Y$, $\overline{Y}$ 代用因子为 $X\overline{Y}$;

$\therefore Z = \overline{X}Y + X\overline{Y} = \overline{X}Y \cdot Y + X\overline{Y} \cdot X = \overline{X}Y \cdot Y \cdot \overline{X}Y \cdot X$ 用此式可得到步骤2中图1.5所示的逻辑图。

实验1.3 组成一个时钟发生器

实验目的

学习用一片555集成电路建立一个时钟发生器。

实验原理

本实验用一片型号为555的集成电路来产生时钟脉冲。555可以产生从1Hz至1MHz频率的时钟信号,但通常最好用在低于300kHz。产生你所需的信号仅须在555芯片外部接两个电阻和一个电容。关于555芯片的有关资料见本书附录B。

实验器材

- TP805 TS或TP80 TS实验板 1 块,
- 示波器 1 台,
- 555时钟发生器 1 片,
- 0.1μF/6V电容 1 个,
- 0.01μF/6V电容 1 个,
- 1kΩ电阻 1 个,
- 56kΩ电阻 1 个。

实验内容和步骤

1. 555时钟发生器接线如图1.7。其中复位开关S可选用面板上的微动开关。
2. 可用下面的公式求得时钟频率(f)和占空比(D)，

$$f = \frac{1.46}{(R_A + 2R_B)C},$$

$$D = \frac{T_H}{T_H + T_L} = \frac{R_A + R_B}{R_A + 2R_B}.$$

3. 若用下列参数，可得到大约1kHz的时钟： $C = 0.01\mu\text{F}$ ， $R_A = 1\text{k}\Omega$ ， $R_B = 56\text{k}\Omega$ ， $f \approx 1.2\text{kHz}$ ， $D \approx 0.5$ 。
4. 用示波器观察电路输出，读出其频率值。

实验1.4 消除开关的抖动

实验目的

学会用双稳电路消除开关(按键)的抖动。

实验器材

- TP805 TS或TP80 TS实验板 1块，
- 74LS00四与非门 1片，
- 4.7k Ω 电阻 2个。

实验内容和步骤

1. 用两个与非门可接成R-S触发器。
2. 现在，R和S均未接任何信号，你认为此时它们(R和S)是什么逻辑状态？
3. 试改变一下触发器的状态：如果LED亮着，请使它变暗；如果LED是暗的，则使它变亮。
4. 将R和S端均通过一个4.7k Ω 电阻接到+5V上，这样要比输入悬空好，为什么？
5. R-S端分别接在一个按钮开关的常闭触点和常开触点上，电路形式如图1.8所示。

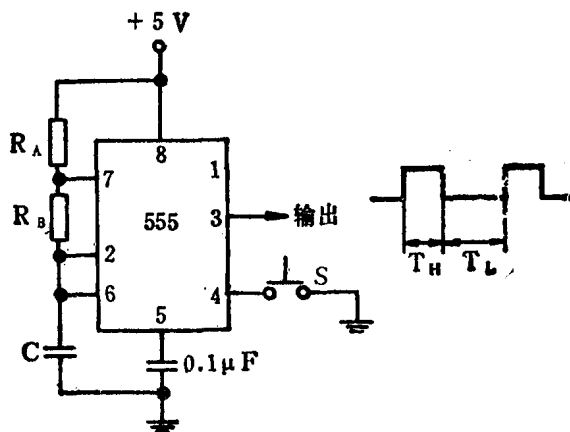


图1.7 555时钟发生器接线图

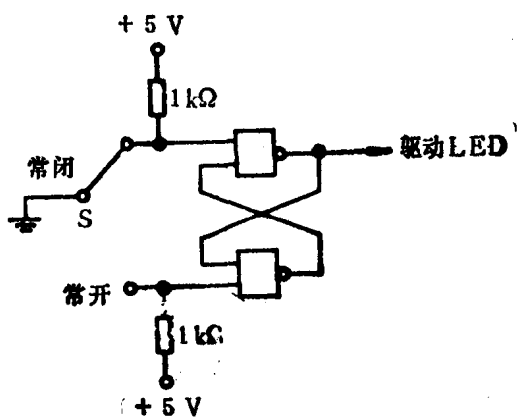


图1.8 防抖动电路

6. 按下S并释放它，观察LED的状态。（在此实验中你可能还不体会开关S的抖动是否被消除。但在以后的实验中你将体会到。若使用示波器观察就很直观了）。

实验二 TP805微型机的操作

实验目的

熟悉键盘操作，熟悉系统监控程序 DEBUGGER及编辑/汇编程序EASM05的使用方法。

实验器材

TP805微型机一台，CRT显示器一台。

实验要求

掌握源程序的输入方法及修改方法，断点的设置、程序的执行以及执行结果的检查。

实验内容和步骤

程序：多字节二进制加法运算

```
                ORG 4000H
START:          LD  B, 5          ; 设两个5字节二进制数相加
                OR   A           ; 清进位位
                LD  DE, 4100H    ; 被加数指针
                LD  HL, 4200H    ; 加数指针
                LD  IX, 4300H    ; 和数指针
ADDMOR:         LD  A, (DE)      ; 被加数1字节送A
                ADC  A, (HL)      ; 与加数带进位加
                LD  (IX), A      ; 结果送内存
                INC  DE          ; 指向下一单元
                INC  HL
                INC  IX
                DJNZ ADDMOR      ; 若B≠0转下一字节相加
                RST 8
                ORG 4100H
                DEFW 3624H
                DEFW 1BA7H
                DEFB 09H
                ORG 4200H
                DEFW 5597H
                DEFW 77BDH
                DEFB 2AH
                ORG 4300H
                DEFS 5
                END START
```

练习2.1 程序的导入和连续执行

1. 开机：先合CRT电源开关，再合主机电源开关；

2. 开机后按 U 键, 进入 USER 状态; 键入 EASM05 则进入编辑/汇编状态, 主机提示:
ERASE TEXT? (Y/N) (要清除文本吗?)

3. 按下 Y, 则屏幕上出现

TP805 EDITOR/ASSEMBLER 1.2

*

及光标闪烁提示符 (注意: 一旦输入源程序后, 再次进入编辑/汇编状态, 若要保留文本时要按 N 键);

4. 用插入命令 I 输入源程序;

5. 在 * 提示符下键入 A-WE, 再按 **RETURN** 键则系统对输入的源程序进行汇编, 生成目的码。如果输入的源程序有错, 系统将给出提示, 此时可用行内编辑命令 E 进行修改;

6. 输入的源程序修改无误后, 再次用 A 命令汇编, 系统将提示:

00000 TOTAL ERRORS

READY CASSETTE

按 **BREAK** 键退出命令级, 机器又出现提示符 *;

7. 打入 M **RETURN**, 则汇编后生成的目的码存入以 4000H 为起始地址的内存中。

注意

如果伪指令 ORG 4000H 误作 OGR 4000, 用汇编命令 A-WE 检查不出错误。在这种情况下, 当打入 M **RETURN** 时系统将提示:

TEXT IS IN DANGER

此时要打 **BREAK** 键, 则系统重新出现 * 提示符等待修改。这是因为 ORG 4000 会将目的码定位在十进制 4000 单元, 对应的十六进制数为 0FA0H 单元, 而小于 2000H 地址的单元为 ROM 区, 故系统不能将目的码写入, 即对 **RETURN** 不予理会, 此时只能用 **BREAK** 或 **ABORT** 键退出。

8. 程序目的码存入内存后, 用 Q **RETURN** 键使系统回到 MANAGER 控制之下;

9. 按 D 键, 在 DEBUGGER 控制下再打入 G 4000 **RETURN**, 则程序连续执行完毕。由于执行最后一条 RST 8H 指令, 系统显示全部 CPU 寄存器内容, 记下 DE、HL 及 PC 中的内容。若程序最后一条指令不是 RST 8H 而是 HALT 指令, 可用 R 命令显示全部 CPU 寄存器内容;

10. 检查运行结果。打入 D 4300 **RETURN** 则屏幕上显示以 4300H 为起始地址到 4377H 共 120 字节内容, 记下运算结果数据。

11. 按下 **BREAK** 键, 退出 D 命令。

问题

退出 D 命令并显示 CPU 寄存器内容还可利用哪个键? 怎样操作?

练习 2.2 单步执行程序

1. 在 DEBUGGER 控制下, 用 M 命令改变被加数和加数数据。操作如下:

打入 M 4100 **RETURN**, 机器显示 4100H 单元中的内容, 键入更改的数据, 按 **RETURN**

键则更改完成, 同时显示下一单元 4101H 内容, 等待修改。全部数据修改完毕时, 用 **BREAK** 键退出 M 命令;

2. 在 DEBUGGER 控制下, 用 X 命令更改 PC 寄存器的内容, 使 PC 中的值为 4000H。操作

如下:

打入X PC 按长键[SPACE],显示 X PC XXXX 键入4000[RETURN]。修改完成后可用R命令或按动一次[ABORT]键来检查PC中内容:

3. 在待命状态下按A键,则执行一条程序指令并自动返回DEBUGGER状态,同时显示全部用户寄存器内容。通过观察寄存器内容,检查是否执行了一条指令;

4. 连续按动A键,直到程序执行完毕;

5. 用D命令检查运算结果。

问题

修改内存中数据,除去用M命令外,还可用什么命令?怎样操作?

练习2.3 设置断点

1. 在重新进入EASM05软件后,不要消除原先的文本,将实验程序改为多字节十进制加法程序。即

(1) 在ADC A,(HL)指令后加一条DAA指令(用I命令插入,注意步长的选取!);

(2) 将被加数、加数改为二十进制数;以上修改过程要求在机器上进行,不要重新键入整个程序。

2. 程序改过以后,在用A命令汇编过程中,按下[RETURN]键或者除[RESET]、[ABORT]以外的任何一键,可使汇编过程停止;再次按动一次[RETURN]键或除[RESET]、[ABORT]以外的任何一键,可使汇编过程继续进行。用此法可找到DJNZ ADDMOR的地址,并记下它,以便在该处设置断点;

3. 用M命令将汇编后的目的代码程序存入内存;

4. 在DEBUGGER控制下,用B命令在DJNZ ADDMOR处设置断点,操作如下:

打入B XXXX[RETURN],其中XXXX即DJNZ指令的地址码;

5. 用B[RETURN]命令检查断点是否已设置好;

6. 用连续执行命令G 4000[RETURN]执行程序,则程序应停在断点处。此时屏幕上显示CPU状态和对应PC值的内存单元的内容。请记下执行一次后各寄存器的内容;

7. 打入G [RETURN],则再执行一次程序,再检查各寄存器的内容,直到执行完毕;

8. 用D命令检查内存中的结果单元,并记下结果数据;

9. 清除断点:键入K [RETURN],则全部断点被清除;

10. 用B命令检查断点是否被消除;

11. 用M命令把4300H为起始地址的5个结果单元通过键盘清零;

12. 用G 4000[RETURN]连续执行程序;

13. 再次用D命令检查结果单元,比较两次执行程序的运算结果,若一致,则证明断点确已被消除;

14. 关机:先关主机电源开关,再关CRT电源开关。

问题

* 在DEBUGGER控制下,若用F命令查找DJNZ指令所在的地址应该怎样操作?有几种方法?

* 用反汇编Z命令可以吗?

实验三 汇编程序的调试

实验目的

学习源程序在机器上的调试方法。

实验器材

TP805一台, CRT一台, 打印机一台。

实验要求

对给出的几个源程序(有的有错误)通过上机调试, 得出正确的结果。

实验内容和步骤

练习3.1 数据块的传送和检索

程序一: 将5000H为首地址的120H字节的数据传送至首地址为6000H的内存中去。

```
HORG 4000
START, LD HL, 5000H ; 首地址
      LD DE, 6000H ; 末地址
      LD BC, 120H ; 字节数
      LDIR
      RST 8H
      END START
```

实验步骤

1. 开机。先合CRT、打印机电源开关, 再合主机电源开关;
2. 输入程序、汇编、存入内存;
3. 用键盘输入的办法, 在5000H—5004H、511BH—511FH键入几个特殊数据如11H、22H、33H、44H、55H等, 以此作为原数据块;
4. 在DEBUGGER状态下, 执行上面的程序一;
5. 用D命令检查6000H—6120H单元数据是否与原数据块相同。

程序二: 数据块比较。数据从5000H复制到6000H后, 为检查复制数据的正确性, 可用下面的程序检查:

```
ORG 4100H
VERIFY, LD HL, 0
      LD (5130H), HL, 将5130H单元清0
      LD HL, 5000H ; 原数据块首地址
      LD DE, 6000H ; 复制的数据块首地址
      LD BC, 120H ; 数据块长度
      LOOP, LD A, (DE) ; 取复制区一字节数据
      CP (HL) ; 与源区对应字节比较
      JR NZ, ERROR; 若不相等(复制出错)转ERROR
      INC DE
      INC HL
```



```

DEC BC
JR NZ, LOOP ; 未检查完, 循环
JR DONE
ERROR, LD (5130H), HL, 将出错时源区相应字节地址送入5130H、5131H单元
DONE, HALT
END VERIFY

```

实验步骤

1. 导入程序二;
2. 执行程序二;
3. 检查5130H、5131H单元内容, 这个结果对吗?
4. 用键盘操作改变复制区611FH单元数据, 即人为制造错误。再执行一遍程序二, 程序能否检查出错在哪里?
5. 找出程序二错在何处, 并改正之;
6. 执行改正过的程序, 若能正确检查出复制块中数据是对还是错, 则程序调试完毕。

问题

你 是通过什么方法找出程序二的错误? 改正过的程序是怎样的? 程序二还能优化吗?

练习3.2 8位二进制带符号数相乘

程序三:

```

; 初始化
ORG 4200H
START, LD A, (DATA1); 取被乘数
LD E, A
LD A, (DATA2); 取乘数
LD HL, 0 ; 清HL, 准备放部分积
LD B, 08 ; 置循环次数

; 检查数据符号, 将负数变正
OR A ; 判乘数符号
JP P, ONE ; 若为正, 转ONE
NEG ; 否则求补变正
ONE, LD C, A ; 变正的乘数暂存C
LD A, E ; A ← 被乘数
OR A ; 判被乘数符号
JP P, TWO ; 被乘数为正转TWO
NEG ; 否则求补变正
TWO, LD E, A ; 变正后的被乘数送E
LD D, 0 ; 被乘数扩为16位数
LD A, C ; 变正后的乘数送A

; 作无符号数乘法
MULT, ADD HL, HL ; 部分积左移一位
RLA ; 乘数左移, 判断位移入标志CY
JR NC, CHCNT; CY=0则转CHCNT
ADD HL, DE ; CY=1, 部分积加被乘数

```