

高等学校计算机系列教材

微计算机技术 ——实验与辅导

仇玉章 冯一兵 编著

清华大学出版社



中国计算机技术丛书

微计算机技术

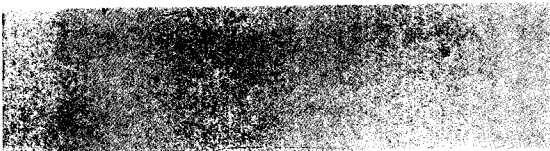
——实践与教学

王明 编著

清华大学出版社



高等学校计算机系列教材



微计算机技术 ——实验与辅导

仇玉章 冯一兵 编著

清华大学出版社
北京

内 容 简 介

本书是《微计算机技术教程》(马群生等编著,清华大学出版社出版)一书的配套教材,也可以作为“汇编语言程序设计”、“微机原理与接口技术”等课程的实验和辅助教材。

本书结合课程内容,从应用的角度出发,针对汇编语言程序设计、接口技术,编排了24个软硬件实验和13个指导性例题。对于各类接口芯片,首先从应用的角度对理论知识进行小结,然后给出指导性例题,最后安排实验。附录中给出每个实验的参考程序,适合进行“微计算机技术”课程教学的师生和从事微机应用的工程技术人员使用。

版权所有,翻印必究。举报电话:010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

本书防伪标签采用特殊防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

微计算机技术——实验与辅导/仇玉章,冯一兵编著. —北京:清华大学出版社,2006.1
(高等学校计算机系列教材)

ISBN 7-302-10321-6

I. 微… II. ①仇… ②冯… III. 微型计算机—高等学校—教学参考资料 IV. TP36

中国版本图书馆 CIP 数据核字(2005)第 002764 号

出 版 者: 清华大学出版社

<http://www.tup.com.cn>

社 总 机: 010-62770175

地 址: 北京清华大学学研大厦

邮 编: 100084

客户服务: 010-62776969

责任编辑: 马瑛珺

印 刷 者: 北京嘉实印刷有限公司

装 订 者: 三河市李旗庄少明装订厂

发 行 者: 新华书店总店北京发行所

开 本: 185×260 印 张: 9.75 字 数: 239 千字

版 次: 2006 年 1 月第 1 版 2006 年 1 月第 1 次印刷

书 号: ISBN 7-302-10321-6/TP·7025

印 数: 1~5000

定 价: 15.00 元

出版说明

清华大学出版社推出的这套《高等学校计算机系列教材》是《清华大学计算机系列教材》的姊妹篇。

《清华大学计算机系列教材》出版以来,多次获得国家 and 部级奖项。我们经常收到一些师生热情洋溢的来信,强烈感受到他们对新的知识与教育模式的渴求,同时也感受到广大师生对清华大学计算学科教学工作的关注和信任。

随着高等教育规模的持续扩大和高等教育改革的不断深入,不同院校对于计算学科的教学工作提出了新的要求,突出体现在:理论课时的压缩、实践能力的要求提高,以及学科教育与行业需求的不断结合。根据这些发展趋势,清华大学一批学术水平高、教学经验丰富的教授总结了他们几十年的教学和科研经验,有针对性地编写了《高等学校计算机系列教材》。这套教材的特点体现在:

1. 课程内容在《清华大学计算机系列教材》的基础上,进行了适时的修订更新,并且明确了教学基本要求,区分应该熟练掌握和只需一般了解的内容。

2. 强调加强基础理论教育,重视学生实践能力的培养。课程内容为进一步的实践教学既提供了基础知识,又留出了足够的时间。

另外,本套丛书同时出版了相关辅导用书,并为教师免费提供电子课件,便于师生的教学使用。

清华大学计算学科坚持推行具有启发性的、富于创造性的教学工作,为国家源源不断地培养出一批又一批优秀人才。从《清华大学计算机系列教材》中就可以体会到这些艰辛的探索历程,希望作为姊妹篇的《高等学校计算机系列教材》也能得到师生的认可。

清华大学出版社

2005年9月

前 言

本书是《微计算机技术》(马群生等编著,清华大学出版社出版)一书的配套教材,也可以作为“汇编语言程序设计”、“微机原理与接口技术”等课程的实验和辅助教材。

本书共分4章,内容如下:

第1章为汇编语言程序实验基础,介绍汇编语言几个应用软件的使用。第2章为汇编语言程序设计,给出指导性例题和汇编语言实验课题。第3章介绍 TPC-1 型 16 位微机实验培训系统的电路设计。第4章为接口实验,首先针对常用接口芯片从应用的角度进行小结,然后给出指导性例题,最后选编实验课题。附录 A 给出了汇编语言编程实验的参考程序。附录 B 给出了接口实验的参考程序。

本书第1章、第2章、第4章、附录 A、附录 B 由仇玉章编写,第3章由冯一兵编写。

由于编者水平所限,难免有错漏之处,敬请读者指正。

作 者

2005 年 5 月

目 录

第 1 章 汇编语言程序实验基础	1
1.1 汇编语言程序的开发过程	1
1.2 汇编语言常用软件的使用方法	2
1.2.1 编辑程序(EDIT. EXE)	2
1.2.2 宏汇编程序(MASM. EXE)	5
1.2.3 连接程序(LINK. EXE)	7
1.2.4 调试程序(CV. EXE)	8
第 2 章 汇编语言程序设计	11
2.1 DOS、BIOS 功能调用	11
2.1.1 DOS 系统 I/O 功能调用	11
2.1.2 BIOS 键盘输入功能调用	13
2.1.3 文本方式 BIOS 屏幕功能调用	14
2.2 汇编语言程序设计举例	16
2.2.1 延时程序	16
2.2.2 字符串动态显示技术	16
2.2.3 菜单程序的设计	20
2.2.4 批处理文件的设计与应用	24
2.3 汇编语言编程实验	27
实验 1 统计负数的个数	27
实验 2 BCD 码数转换成二进制数显示	27
实验 3 计算平均成绩	28
实验 4 通行字识别	28
实验 5 分类统计单科成绩	29
第 3 章 TPC-1 型 16 位微机实验培训系统	31
3.1 概述	31
3.2 公共控制电路	31
3.3 辅助电路	33
3.4 接口实验电路	35

第4章 接口实验	41
4.1 8段数码管电路	41
4.1.1 数码管电路工作原理	41
4.1.2 数码管电路实验	44
实验1 1位数码管闪亮显示	44
实验2 2位数码管交替显示	44
实验3 6位数码管静态/动态混合显示	44
实验4 6位数码管动态显示	45
4.2 可编程定时器与定时中断	46
4.2.1 8254/8253 应用小结	46
4.2.2 PC机音乐程序设计	51
4.2.3 8254 计数过程通用测试程序	56
4.2.4 定时中断程序设计举例	60
4.2.5 可编程定时器与定时中断实验	65
实验1 中断方式的数码管交替显示	65
实验2 定时中断屏显字符串	65
实验3 中断方式的数码管动态显示	66
实验4 简易菜单选择的音乐程序	67
实验5 菜单选择的音乐程序	67
4.3 并行接口	68
4.3.1 8255A 应用小结	68
4.3.2 8255A 双向通信举例	72
4.3.3 8255A 驱动并行打印机	76
4.3.4 模拟交通灯控制	79
4.3.5 8255A 并行接口实验	83
实验1 8255A 选通型输出实验	83
实验2 小键盘按键识别数码管显示	84
实验3 小键盘按键识别屏幕显示	84
实验4 小键盘按键识别数码管移位显示	85
4.4 串行异步通信	86
4.4.1 串行异步通信小结	86
4.4.2 串行异步通信实验	94
实验1 PC机串行口内环自动测试	94
实验2 PC机串行口自动测试	94
实验3 8251A 异步通信	95
4.5 数模、模数转换	97
4.5.1 数模转换实验举例	97

4.5.2	模数转换实验举例	99
4.5.3	数模、模数转换实验	102
实验 1	数模转换产生正弦波	102
实验 2	3 路模数转换	102
实验 3	3 路模数转换多窗口显示	103
附录 A	汇编语言编程实验参考程序	105
附录 B	接口实验参考程序	112

第 1 章 汇编语言程序实验基础

1.1 汇编语言程序的开发过程

汇编语言程序的开发过程如图 1.1 所示。这个过程主要由编辑、汇编、连接几个步骤构成。

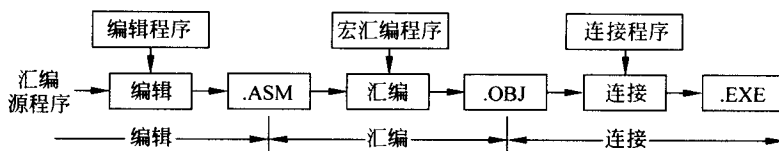


图 1.1 汇编语言程序的开发过程

1. 源程序的编辑

编辑过程就是调用编辑程序把源程序输入内存,生成一个扩展名为 ASM 的文本源文件并存入磁盘。如果是对原有的 ASM 文件进行修改,还会自动生成一个扩展名为 BAK 的备份文件,它保存的是修改前的 ASM 文件。用 DOS 提供的 EDIT. EXE 或其他全屏幕编辑软件都能完成编辑任务。

2. 源程序的汇编

汇编就是调用汇编程序(如 MASM 或 TASM)对源程序进行翻译,生成扩展名为 OBJ 的目标文件。在汇编过程中,若汇编器检查到源程序中有语法错误,则不生成目标文件,但给出错误提示信息。根据用户需要,汇编程序还可生成列表文件(LST 文件)和交叉引用文件(CRF 文件)。

3. 目标程序的连接

连接的过程是:调用连接程序(如 LINK 或 TLINK)将用户目标程序和库文件进行连接、定位,生成扩展名为 EXE 的可执行文件。连接时,如果在目标文件或库文件中找不到所需的连接信息,则连接程序给出错误提示信息,而不生成可执行文件。根据用户需要,连接程序还可以生成内存分配文件(MAP 文件)。

4. COM 文件的生成

按照 COM 文件的汇编格式设计的源程序,在生成 EXE 文件之后,还要转换成 COM 文件。如果连接时使用 TLINK. EXE 程序,在 TLINK 命令后加选项“/t”,可直接生成 COM 文件。但是宏汇编 MASK. EXE(5.0 版本以下)不支持此项功能,还需要调用转换程序 EXE2BIN. EXE,才能将 EXE 文件转换成 COM 文件。假设用户的 EXE 文件、COM 文件

和转换程序都在同一目录下,则键入:

EXE2BIN "文件名".EXE "文件名".COM

如果不能转换就给出错误提示信息。

5. 调试可执行程序

可执行程序运行之后没有得到预想的结果,就需要重新审查源程序,找出算法或者逻辑错误,再重新编辑、汇编、连接和执行。在调试程序的时候,也可以借助调试软件。

1.2 汇编语言常用软件的使用方法

建议读者在 C(或 D)盘建立一个工作子目录,将汇编语言常用的软件装入工作子目录之下,在工作子目录下,完成编辑、汇编和连接等操作,生成的用户文件也存放在工作子目录之下。本节在介绍常用软件使用方法时,不再说明文件的盘符和路径。

1.2.1 编辑程序(EDIT.EXE)

1. 启动 EDIT 进入编辑状态

(1) 在 DOS 环境下键入: EDIT ✓

这条命令执行 EDIT.EXE 程序,自动进入编辑状态,屏幕顶部显示主菜单,底部行显示提示信息,中间的 22 行为编辑窗口,用户使用编辑命令逐行输入源程序。由于该条命令没有给出待编辑的文件名,因此编辑后文件存盘时,要键入“Alt+F”,选择其中的 Save 功能,并通过会话给出文件名,才能将编辑后的文件存盘。

(2) 在 DOS 环境下键入: EDIT 待编辑的文件名.ASM ✓

例如,键入: EDIT ABC.ASM ✓

这条命令有 3 个作用:

- ① 执行 EDIT.EXE 程序,自动进入编辑状态,屏幕顶部显示主菜单。
- ② 如果工作目录中没有“ABC.ASM”文件,则编辑窗口没有信息,编辑后的文件存盘时,将自动存放在 ABC.ASM 文件中。
- ③ 如果工作目录中有“ABC.ASM”文件,系统将自动把 ABC.ASM 文件调入内存,并将开始的 22 行语句显示在编辑窗口供编辑。

2. 常用的编辑命令

Page Up	屏显上一页语句行
Page Down	屏显下一页语句行
↑、↓、←、→	光标上移、下移一行,左移、右移一位
Ctrl + ↑	屏幕信息向下移动一行
Ctrl + ↓	屏幕信息向上移动一行
Ctrl + ←	光标向左移动一个区段
Ctrl + →	光标向右移动一个区段

Ctrl + Home	光标移到文件的开始行
Ctrl + End	光标移到文件的末尾
Home	光标移到当前行的开始
End	光标移到当前行的末尾
Enter	插入新的一行
Delete	删除光标所在位置上的一个字符
Backspace	删除光标左侧的一个字符
Ctrl+Y	删除光标所在行的全部信息

说明:

Ctrl+Y 是双键操作,即按下 Ctrl 不松手,再按 Y 键。其他的双键操作类同。

3. EDIT 的编辑功能

EDIT 采用两级下拉式菜单显示编辑功能。启动 EDIT 之后,屏幕顶部显示 6 个主菜单项,键入“Alt + 主菜单项首字母”即能显示相应的二级菜单;或者先按 Alt 键,然后按“←”、“→”键移动光条,再按回车键,也能显示二级菜单。显示二级菜单之后,按“↑”、“↓”键移动光条,再按回车键,即可选择相应的编辑功能;也可以通过热键操作选择编辑功能。本小节仅作简要介绍,对于比较复杂的编辑功能,下文再作专题描述。

(1) File 这项功能的二级菜单提供了与文件有关的操作。

- New 建立一个新文件。
- Open 从磁盘装入一个文件。
- Save 当前编辑的文件存盘。
- Save As 改用新的文件名存储当前文件。
- Print 打印文件。
- Exit 退出编辑状态,返回 DOS。

(2) Edit 对文件进行“块操作”。

(3) Search 完成字符串搜索与替换。

(4) View 进行多窗口编辑。

(5) Options 它有两个子功能。其一,设置 Tab 键控制的光标移动格数;其二,设置打印口。

(6) Help 显示帮助信息。

4. 块操作功能

若干连续的语句行称为一个“程序块”,一行中若干连续的字符称为一个“字符块”。主菜单 Edit 提供了“块定义”、“块复制”、“块转移”、“块删除”操作,尤其是块复制、块转移,为修改源程序提供了很大的方便。进行块复制、块转移、块删除之前,必须进行块定义。假设源程序的第 10~15 行为一个程序块,下面以此为例,介绍块操作的方法。

(1) 块定义操作步骤

- ① 按“↑”、“↓”键将光标移动到第 10 行。
- ② 按下 Shift 键(不松手),再连续按“↓”键直到第 15 行,光标经过的行将变色,显示为

白底黑字,从而完成了“程序块”的定义。如果要定义字符块,应先将光标移动到字符块的首字符,然后按下 Shift 和“→”键,即可完成“字符块”的定义。

(2) 块复制操作步骤

假设把源程序的第 10~15 行复制一份到第 20~25 行。

① 首先完成第 10~15 行的块定义。

② 键入“Ctrl+C”,将定义的程序块复制到 EDIT 指定的缓冲区。

③ 移动光标到第 20 行。

④ 键入“Ctrl+V”,将缓冲区中的程序块复制一份到第 20~25 行,就完成了“块复制”的操作。缓冲区中的程序块并不消失。

(3) 块转移操作步骤

假设把源程序中的第 10~15 行转移到第 20~25 行。

① 首先完成第 10~15 行的块定义。

② 键入“Ctrl+X”,将定义的程序块复制到 EDIT 指定的缓冲区,并且删除被定义的程序块。

③ 移动光标到第 20 行。

④ 键入“Ctrl+V”,将缓冲区中的程序块复制一份到第 20~25 行,就完成了“块转移”的操作。缓冲区中的程序块并不消失。

(4) 块删除操作步骤

首先进行块定义操作,然后按 Delete 键,被定义的程序块即被删除。

5. 多窗口编辑功能

当程序员正在编辑某个程序,同时又想参考另一个程序的时候,怎么办呢?主菜单 View 的多窗口编辑功能可以同时把两个程序调入编辑窗口,供程序员查阅。多窗口的编辑功能还可以实现文件之间的块复制、块转移。虽然如此,但读者应该慎重使用文件之间的块转移操作,其原因是不言而喻的。

(1) 进入多窗口编辑环境

假设正在编辑的程序是“1. ASM”,需要参考的程序是“2. ASM”。怎样进行多窗口编辑呢?步骤如下:

① 首先将“1. ASM”调入编辑窗口。

② 选择 View|Split Window,该项功能把编辑窗口一分为二,形成上、下两个窗口。

③ 选择 File|Open,将“2. ASM”程序装入内存。此时,上窗口显示的是“1. ASM”的语句行,下窗口显示“2. ASM”的语句行,按“↑”、“↓”键可以使下窗口程序滚动显示,从而达到查阅“2. ASM”程序的目的。

(2) 文件之间的块复制操作

在上述操作的基础上,怎样把“2. ASM”程序中的第 10~15 行语句复制到“1. ASM”程序的第 20~25 行呢?步骤如下:

① 按“↑”、“↓”键,将光标移动到下窗口“2. ASM”程序的第 10 行。

② 按下 Shift 键不松手,再按下“↓”键,完成第 10~15 行语句的“块定义”。

③ 键入“Ctrl+C”,将定义的程序块复制到 EDIT 指定的缓冲区。

④ 选择 View|Close Window, 这项功能将取消多窗口显示, 使编辑窗口恢复, 全屏幕显示“1. ASM”程序的语句行。

⑤ 将光标移动到“1. ASM”程序的第 20 行。

⑥ 键入“Ctrl+ V”, 将缓冲区中的语句行复制到“1. ASM”程序的第 20~25 行, 完成文件之间的“块复制”。

6. 字符串搜索与替换

变量、标号、寄存器名、立即数等都是字符串。在一个大型程序中, 如果某字符串被多次引用, 现在需要用另一个字符串去替换它, 在这种情况下, 由于语句行比较多, 人工搜索与替换比较费事, 而且容易遗漏, 此时可借助“字符串搜索与替换”功能。主菜单 Search 提供的字符串搜索与替换功能, 有两种搜索方式: 一种是单步搜索, 另外一种连续搜索。操作步骤如下:

① 按“Ctrl+Home”键, 将光标移到文件的首部。

② 选择 Search|Replace, 屏幕上立即出现一个对话框。在 Find what 一栏中键入要搜索的字符串, 在 Replace with 一栏中键入用以替换的字符串。接下来选择搜索方式。

③ 用户将光标移动到 Replace 框, 按下回车键之后, EDIT 自动进行单步搜索, 每搜索到一个匹配的字符串, 就停止搜索, 屏幕显示另一个对话框, 询问用户当前搜索到的字符串是否要替换它。用户把光标移到 Replace 框, 按回车键, 就实现替换功能; 反之把光标移到 Skip 框按回车键, 就是不替换。

④ 用户将光标移动到 Replace All 框, 按下回车键之后, EDIT 将连续搜索并自动替换, 不再征询用户意见。

1.2.2 宏汇编程序(MASM. EXE)

宏汇编程序以源程序(一定要有扩展名 ASM)为汇编对象。汇编过程中, 宏汇编程序对源程序进行两次扫描, 检查其语法错误, 如果没有语法错误, 则根据程序员的要求生成目标文件(OBJ)、列表文件(LST)和交叉引用文件(CRF)。宏汇编程序启动后, 首先显示版本信息, 然后逐条进行人机会话。

宏汇编程序可以生成文件名互不相同的 OBJ 文件、LST 文件和 CRF 文件, 但这样做没有好处, 一般都生成与源文件同名的目标文件(OBJ)。假设待汇编的源程序为“ABC. ASM”, 并以此为例介绍人机会话过程。

1. 启动宏汇编程序的 3 种方法

(1) 在 DOS 环境下键入: MASM ✓

系统会给出下列机上提示

人机会话过程如下:

Source filename[. ASM]: ABC ✓

; 键入源程序名

Object filename[ABC. OBJ]: ✓

; 生成的目标文件为 ABC. OBJ

Source listing[NUL. LST]: ✓

; 不生成 LST 文件

Cross_reference[NUL.CRF]: ✓ ; 不生成 CRF 文件

ABC.ASM(10): error A2009: Symbol not defined: MESH

0 Warning Errors ; 0 个警告性错误

1 Severe Errors ; 1 个严重错误

这种启动方式有 4 句会话。

第 1 句: 请键入待汇编的源程序名。

第 2 句: 询问生成的目标文件是否是 ABC.OBJ, 按回车键表示认可。

第 3 句: 询问是否要生成列表文件, 按回车键表示不生成 LST 文件。如果要生成 LST 文件, 请键入文件名。

第 4 句: 询问是否要生成交叉引用文件, 按回车键表示不生成 CRF 文件。

会话结束, 宏汇编程序开始汇编, 并给出汇编结果。上述源程序汇编后, 有 1 个严重错误, 即 ABC.ASM 程序的第 10 行, 有 1 个符号名 MESH 没有定义过。程序员应当重新编辑, 改正错误, 然后再次汇编, 直到没有语法错误为止。假设语法错误已经改正了。

(2) 在 DOS 环境下键入: MASM 待汇编的源文件名 ✓

例如: MASM ABC ✓

这种启动方式在命令行中给出待汇编的源文件名, 因此人机会话只有 3 项内容, 示范如下:

Object filename[ABC.OBJ]: ✓ ; 生成的目标文件为 ABC.OBJ

Source listing[NUL.LST]: ✓ ; 不生成 LST 文件

Cross_reference[NUL.CRF]: ✓ ; 不生成 CRF 文件

0 Warning Errors ; 0 个警告性错误

0 Severe Errors ; 0 个严重错误

(3) 在 DOS 环境下键入: MASM 待汇编的源文件名; ✓

例如: MASM ABC; ✓

这种启动方式在命令行中给出待汇编的源文件名, 并且以分号结束。按下回车键之后, 没有人机会话过程, 汇编结束后, 同样给出汇编信息。如果源程序没有语法错误, 则自动生成同名的 OBJ 文件, 不生成 LST 文件和 CRF 文件。推荐使用这种方法。

2. 宏汇编参数

启动宏汇编程序的时候, 可以携带参数。宏汇编参数是任选项, 参数以“/”开头, 紧跟在 MASM 之后, 参数可以连用, 在 DOS 环境下键入“MASM/H ✓”之后, 屏幕上就能显示该版本宏汇编程序的参数表和各参数的功能。下面介绍几个比较有用的宏汇编参数:

/L 汇编后能生成 LST 文件。

/Z 如果源程序有错, 汇编结束后, 能显示有语法错误的语句行。

/Zi 产生用于 Code View 的符号信息。

1.2.3 连接程序(LINK. EXE)

经过汇编之后生成的目标文件(OBJ)不能在 PC 上运行,必须经过连接程序的连接和定位才能生成可执行文件(EXE)。

对于多模块程序(即源程序由 1 个主模块和若干子模块组成),汇编后生成的是各个模块的目标文件,连接程序要把各个模块的目标文件连接在一起,才能生成可执行文件。

如果设计源程序的时候,调用了库文件子程序(库文件由程序员自己创建,它是若干子程序的集合,文件扩展名为 LIB),连接程序还要把被调用的库文件子程序和目标文件连接在一起才能生成可执行文件。

主教材和实验教程中的例题、实验均为单模块程序,而且没有库文件,因此在启动连接程序,回答问句“Libraries[. LIB]”的时候,请按下回车键,表示没有库文件。

启动连接程序也有 3 种方法,假设目标文件为 ABC. OBJ,并以此为例介绍连接程序启动后的人机会话过程。

(1) 在 DOS 环境下键入: LINK ✓

人机会话过程如下:

Object Modules[. OBJ]: ABC ✓	; 键入目标文件名
Run File[ABC. EXE]: ✓	; 生成的可执行文件为 ABC. EXE
List File[NUL. MAP]: ✓	; 不生成 MAP 文件
Libraries[. LIB]: ✓	; 没有库文件参与连接

LINK: Warning L4021: no stack segment

说明:

① 扩展名 MAP 是映像文件,它给出程序中每个逻辑段的位置和段长度,一般不需要生成 MAP 文件,因此在回答问句“List File[NUL. MAP]”时,按下回车键即可。

② 如果连接时发现错误,连接程序将会给出错误提示信息。

③ 连接程序要求生成 EXE 文件的源程序格式中,必须有堆栈段(堆栈段的标志是该逻辑段的段定义语句中有“STACK”段参数),否则连接之后,连接程序将给出“没有堆栈段”的警告信息。但是,这不代表源程序有语法错误。在程序设计中,如果没有大批数据进出堆栈,为了简化程序设计,程序员往往不单独设置堆栈段,DOS 在装载没有堆栈段的 EXE 文件时,将会自动给程序分配至少 128B 的堆栈区。

(2) 在 DOS 环境下键入: LINK 目标文件名 ✓

例如: LINK ABC ✓

这种启动方式在命令行中直接给出待连接的目标文件名,按下回车键之后,只进行后 3 项人机会话。

(3) 在 DOS 环境下键入: LINK 待连接的目标文件名; ✓

例如: LINK ABC; ✓

这种启动方式在命令行中给出待连接的目标文件名,并以分号结束,按下回车键之后,没有人机会话过程。如果程序没有错误,连接后自动生成同名的 EXE 文件,不生成 MAP 文件。如果没有库文件参与连接,这种方法是比较快捷的。

生成可执行文件(EXE)之后,在 DOS 环境下,键入其文件名即可执行了。必须指出,没有汇编错误,没有连接错误,只能说明源程序没有语法错误;如果源程序在算法或者其他方面有错,那么执行结果也是不正确的,此时应认真分析,找出原因,再重新编辑、汇编和连接。

1.2.4 调试程序(CV. EXE)

CV 是 Code View 的缩写,CV. EXE 是 MASM 5.1 软件包中的一个调试程序,它能够显示待调试的源程序及其目标代码。在 Code View 控制下,可以慢速地或者单步运行程序,监视程序的转向,可以实时地观察相关寄存器内容的变化,也可以检查和修改内存单元的数据,本小节仅介绍 Code View 用于调试汇编语言程序的常用命令。假设 CV. EXE 和待调试的程序(假设为 ABC. EXE)都在同一目录下。

(1) 启动 Code View 的命令

在 DOS 环境下键入: CV 待调试的文件名 ✓

例如: CV ABC ✓

Code View 启动后,屏幕显示如图 1.2 所示。

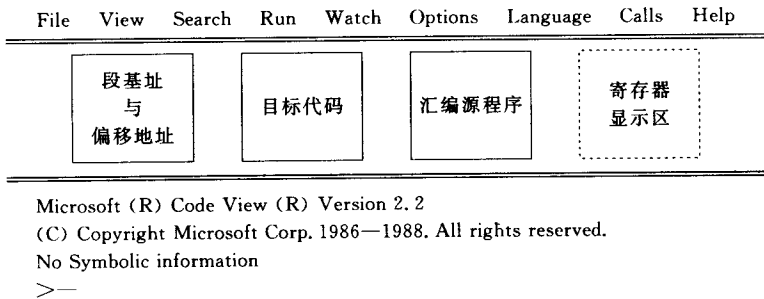


图 1.2 Code View 启动后的屏幕显示

Code View 采用两级下拉式菜单,顶部为 9 个主菜单项。键入“Alt + 主菜单项首字母”可以显示属下的二级菜单,或者先按下 Alt,再按“←”、“→”键,移动蓝色小光条使之覆盖某一菜单项,然后再按回车键,也能显示其属下的二级菜单。二级菜单项显示后,按“↑”、“↓”键移动蓝色小光条,再按回车键即可选择相应的菜单项功能。按下 Esc 键可以关闭二级菜单或者当前的显示框。

主菜单以下,两条双线框之内是调试显示窗口,显示待调试程序的前 18 行语句和相应的目标代码,一条长长的蓝色光条覆盖程序的启动指令。

双线框以下为命令窗口,显示 Code View 的版本和版权说明。“>”号是 Code View 的提示符,“—”是闪烁的光标。

(2) 扩大/缩小显示窗口的命令: Ctrl+T/Ctrl+G

键入“Ctrl+T”可以扩大显示窗口,使其最多能显示 21 行指令;键入“Ctrl+G”,则减小显示窗口的面积。

(3) 光标移动命令: F6

按 F6 键可以使光标在上下窗口之间移动,当光标移到上窗口时,再按“↑”、“↓”键可以使光标移动到某一程序行,按 PgUp 或 PgDn 键可以换页显示。

(4) 打开/关闭寄存器显示窗口的命令: F2