

IBM PC

TURBO PASCAL

程序设计手册

张 帆 李晓林 编著

湖南科学技术出版社

IBM PC

TURBO PASCAL

程序设计手册

张 帆 李晓林编著

湖南科学技术出版社

IBM PC TURBO PASCAL

程序设计手册

张 帆 李晓林编著

责任编辑：古 华

•

湖南科学技术出版社出版发行

(长沙市展览馆路8号)

湖南省新华书店经销 湖南省新华印刷二厂印刷

•

1988年6月第1版 1989年12月第2次印刷

开本：787×1092毫米 1/16 印张：15.25 字数：379,000

印数：5,001—8,000

ISBN 7—5357—0340—2

TP·10 定价：5.50元

地科89—39

内 容 提 要

由于 Turbo Pascal 系统的功能远远超过了以前的各种 Pascal 系统,可以预期,Turbo Pascal 即将成为最流行的程序设计工具之一。本书正是为适应这一需要而编写的。书中列举了大量的编程实例,介绍了许多编程的关键技术,详细阐述了 Turbo Pascal 的基本元素和编程步骤。本书既可作为 Pascal 语言学习的教材,又可作为 Turbo Pascal 程序设计的技术手册,是广大从事计算机学习和科研人员的必备工具书。

目 录

引论	(1)	§ 5.3 无类型文件	(65)
第一章 Turbo Pascal程序设计初步	(2)	§ 5.4 I/O检查	(67)
§ 1.1 运行第一个程序	(2)	第六章 指针	(70)
§ 1.2 系统配带的文件	(4)	§ 6.1 定义指针变量	(70)
§ 1.3 安装系统	(5)	§ 6.2 与指针有关的过程和函数	(70)
§ 1.4 编辑、编译和运行	(6)	§ 6.3 使用指针	(72)
§ 1.5 Turbo编辑器	(11)	第七章 过程和函数	(75)
第二章 程序的基本元素	(19)	§ 7.1 参数	(75)
§ 2.1 符号、保留字和标准标识符	(19)	§ 7.2 过程	(77)
§ 2.2 标准、纯量和子界数据类型	(21)	§ 7.3 标准过程	(78)
§ 2.3 标识符、常数、注释和编译开关	(23)	§ 7.4 函数	(80)
§ 2.4 程序结构	(26)	§ 7.5 标准函数	(81)
第三章 表达式和语句	(28)	§ 7.6 向前引用	(84)
§ 3.1 表达式	(28)	第八章 编制大程序的关键技术	(86)
§ 3.2 赋值语句	(30)	§ 8.1 拼接源程序	(86)
§ 3.3 过程语句	(31)	§ 8.2 覆盖系统	(91)
§ 3.4 Goto语句	(31)	§ 8.3 Chain过程和Execute过程	(97)
§ 3.5 空语句	(31)	§ 8.4 Dos系统功能调用	(100)
§ 3.6 复合语句	(32)	§ 8.5 Dos系统中断调用	(101)
§ 3.7 If语句	(32)	§ 8.6 调用外部子程序	(103)
§ 3.8 Case 语句	(33)	§ 8.7 嵌入机器代码	(105)
§ 3.9 For语句	(33)	第九章 彩色、图形、窗口和声音	(107)
§ 3.10 While语句	(34)	§ 9.1 屏幕模式控制	(107)
§ 3.11 Repeat语句	(34)	§ 9.1.1 文本模式	(107)
第四章 结构数据类型	(35)	§ 9.1.2 图形模式	(108)
§ 4.1 字符串类型	(35)	§ 9.2 窗口	(110)
§ 4.2 数组类型	(38)	§ 9.2.1 文本窗口	(110)
§ 4.3 记录类型与With语句	(39)	§ 9.2.2 图形窗口	(111)
§ 4.4 集合类型	(42)	§ 9.3 基本图形	(114)
§ 4.5 类型常数	(44)	§ 9.4 扩展图形	(118)
第五章 文件系统	(47)	§ 9.5 海龟作图	(124)
§ 5.1 有类型文件	(47)	§ 9.6 声音	(131)
§ 5.2 文本文件	(52)	第十章 其它功能	(138)
		§ 10.1 目录管理	(138)
		§ 10.2 绝对地址变量和内存数组	(140)

§ 10.3	绝对地址函数	(141)
§ 10.4	使用I/O端口	(142)
§ 10.5	编写I/O驱动程序	(142)
§ 10.6	编写错误处理程序	(144)

第十一章 内部数据格式及内存管理

		(145)
§ 11.1	基本数据类型	(145)
§ 11.2	结构数据	(146)
§ 11.3	参数	(148)
§ 11.4	函数结果	(149)
§ 11.5	堆和栈	(149)
§ 11.6	内存管理	(149)

第十二章 程序设计实例：全屏幕汉字编辑程序

		(151)
§ 12.1	功能、设计思想及实现技巧	(151)
§ 12.2	主要过程与函数	(152)
§ 12.3	源程序	(155)

第十三章 Turbo-87、TurboBCD和

汉化Turbo Pascal	(178)
----------------	-------

§ 13.1	Turbo-87 Pascal	(178)
§ 13.2	TurboBCD Pascal	(185)
§ 13.3	用Turbo Pascal处理汉字信息	(190)

附 录

附录A	标准过程与函数一览表	(192)
附录B	运算符一览表	(198)
附录C	编译开关	(200)
附录D	Turbo与标准Pascal	(202)
附录E	编译错误信息	(203)
附录F	运行时的错误信息	(208)
附录G	I/O错误信息	(209)
附录H	改写错误信息	(211)
附录I	Turbo Pascal语法	(215)
附录J	ASCII码表	(219)
附录K	键盘返回码	(220)
附录L	系统中断调用一览表	(223)
附录M	系统功能调用一览表	(229)
附录N	HELP!!!	(235)

引 论

Pascal是一种通用的高级程序设计语言，它是1968年由瑞士苏黎士联邦工业大学的N·沃尔斯(Niklaus Wirth)教授创立的，并为纪念法国十七世纪著名哲学家和数学家布莱斯·帕斯卡(Blaise Pascal)而取名为Pascal。

沃尔斯教授定义的Pascal语言出版于1971年(发表在瑞士的“ETH”杂志上)，用之帮助系统地研究计算机程序设计的教学，尤其引人注目的是其中引入了结构程序设计思想。从此，几乎在任何计算机上的几乎所有的任务都用Pascal来编程。今天，无论在教学还是在专业程序设计方面，Pascal都成了第一流的高级程序设计语言之一。而Turbo Pascal系统可满足各方面用户的需求：即提供给用户一个极其友好的交互式环境；可大幅度提高编程效率，是程序设计的一个极其有效的软件开发工具。

Turbo Pascal严格地遵守由K·Jenson和N·Wirth在Pascal用户手册及报告中定义的标准Pascal。我们在附录D中描述了它们之间的极小差异，然而Turbo Pascal增加了许多必要且能够为编程提供方便的设施和功能，其中有：

- 内存和·COM文件两种执行方式
- 编译和运行发现错误后，自动转入编辑状态
- 绝对地址变量
- 直接存取主机内存和I/O端口
- 动态字符串
- 说明部分的各节可以自由排列
- 可直接使用操作系统调用和中断调用
- 程序内部嵌入机器代码
- 嵌入文件
- 可对整型量进行逻辑运算
- 覆盖系统
- 带有公共变量的程序链接
- 对汇编程序模块的调用
- 随机存取数据文件
- 结构常数
- 类型转换函数
- 完全支持树型结构的目录管理和文件管理
- 可编错误处理例程
- 可编I/O驱动例程
- 彩色功能
- 图形功能
- 海龟作图功能
- 窗口功能
- 音响功能

另外，还增加了许多实用的函数和过程。

该系统提供了三套编译系统：

(1) Turbo——生成8088CPU目标代码。

(2) Turbo-87——生成支持8037协处理器浮点运算的代码，可以提高实数运算的速度与精度。

(3) Turbo BCD——生成支持BCD浮点格式的8088代码，可以处理更高精度的实数运算。

另外，国内还推出了汉化的Turbo版本——CTurbo。

本书着重介绍IBM PC Turbo Pascal的功能及其使用方法，并适当给出一些例子。在最后一章介绍了用Turbo Pascal实现的一个全屏幕汉字编辑器的程序实例。

第一章 Turbo Pascal程序设计初步

本章首先通过介绍一个简单程序的实践,使读者初步了解Turbo Pascal程序编制的步骤,第二节介绍Turbo Pascal 系统盘上所配带的所有文件及其用途,第三节介绍如何用 Tinst 为 Turbo Pascal配置所需环境,第四节介绍Turbo Pascal 的主菜单命令,第五节介绍其编辑命令。

§ 1.1 运行第一个程序

学习一门语言系统的最好办法是首先运行一个简单的程序。这一节给出一个小示例程序,读者不妨照所列出的步骤在机器上编制、运行这个程序。如果你从未学过Pascal语言,那么这一节将使你能够编Pascal程序,如果你已学过 Pascal语言,通过这一节,你便可以运行你以前所编的任何Pascal程序了。

第一步:编写一个程序。程序如下:

```
Program my__first__Exercise,  
{This is my first program}  
Const  
    N = 10;  
Var  
    s, i : integer;  
begin  
    Writeln ('N = ', N);  
    s := 0;  
    For i := 1 to N do s := s + i;  
    Writeln ('1 + 2 + ... + N = ', s);  
end.
```

Pascal程序由程序头,说明部分和执行部分三部分组成。

第一行称为程序头。保留字Program标志着程序的开头,后面跟着程序名my__first__Exercise。

说明部分用来定义程序中要使用的变量和常数。Const N = 10定义了一个整型常数N,其值为10。Var s, i: integer定义了两个整型变量,名字取为s和i。

begin和end这两个保留字的中间部份为该程序的执行部分。程序首先从它的第一句 Writeln ('N = ', N) 开始执行,这是一个输出过程语句,它输出字符串'N = '和整数N的值(即10), s := 0是一个赋值语句,它将整型变量s赋值为0。For语句是一个循环语句,程序中该语句的执行结果是将1, 2, 直至N(即10)连加起来,存在变量s之中。第二个 Writeln

过程语句, 输出 ' $1+2+\dots+N=$ ' 和 s 的值, 也即 $1+2+\dots+N$ 的结果。

程序中, 每个语句之间用分号“;”分隔。由{ }括起来的文字是注解, 对程序的运行没有任何影响。例如: {This is my first program}是一个注解。

第二步: 启动Turbo Pascal系统。将Turbo Pascal系统盘插入当前驱动器, 键入命令TURBO)。Turbo系统被调入运行后提问:

Include error messages (Y/N)?

请键入Y。接着系统显示其主菜单如下:

```
Logged drive: A
Active directory:
Work file:
Main file:
Edit Compile Run Save
Dir Quit compiler Options
Text: 0 bytes
Free: 63484 bytes
```

,

第三步: 将程序装入计算机。在主菜单下, 键入E(编辑命令), 系统提问:

Work file name:

即请输入存储程序的文件名, 文件名可以随便给定, 在此你可以回答FIRST作为存储程序my__first__exercise的文件名, 系统将自动为这个文件加上一个后缀.pas, 文件名变为 FIRST.pas。

回答了文件名后, 系统进入了编辑状态。在这状态下你可逐字逐行输入程序my__first__exercise, 在输入过程中可以使用如下编辑键:

Ins——插入开关。按Ins键实现插入状态与替换状态的转换。在替换状态下, 每敲入一个字符, 被写在光标处, 在插入状态下, 敲入的字符被插入在光标处;

^G——删除光标位置处的字符;

↑、↓、←、→——控制上、下、左、右移动光标;

↵——回车键, 使光标跳至下一行之首;

^Y——删除一行;

^N——插入一行。

其中^Y表示控制键<ctrl>与Y键同时按下。

整个程序输入后, 用^KD键退回到主菜单, ^KD表示先同时按控制键<ctrl>与K键, 再按D键。

第四步: 编译。当你退回到主菜单后, 键入命令C便开始编译你刚输入的程序, 数秒钟后, 编译工作便完成。若在编译中, 系统发现源程序有错, 将显示错误信息, 待键入<ESC>键后, 自动进入编辑状态, 并将光标停留在出错位置上, 修改好错误后, 以^KD退出, 再按命令C重新编译, 直至编译正确无误为止。

第五步: 运行。程序编译完成后, 键入命令R, 则运行所编程序, 程序运行过程中会显示出如下信息:

N = 10

$1+2+\dots+N=55$

第六步：保存程序并退出Turbo系统。键入命令S，使所编程序存入到磁盘中，再键入命令Q，退出Turbo系统。

至此，我们便初步了解了编写、输入和运行一个程序的全过程。

§ 1.2 系统配带的文件

Turbo Pascal(3.0版)系统盘包含如下文件(其中，扩展名为.COM的文件是可执行文件，为.Pas的文件为Pascal源程序)。

TURBO.COM Turbo Pascal系统程序，包括编辑器、编译器等。在DOS系统下键入命令TURBO后，便可装入并运行之。

TURBO.MSG 包含错误信息的文件。如果不需要解释编译时的错误，它可以不在当前盘上，那么编译出错时只显示错误码。可以在附录E中查到其解释。若不将该文件装入内存，可节省约1.5k内存。该错误信息文件还可改写为其它语言(比如中文)，其格式和改写方法请见附录H。

TINST.COM 安装程序。当键入TINST后，运行一个全部由菜单引导的安装程序，它实现三个方面的安装：(1)终端配置；(2)编辑键的定义；(3)指定TURBO.MSG的路径名。该程序不必放在当前盘上。

TINST.MSG 安装程序的信息文件。

TURBO-87.COM Turbo Pascal系统的8087版本。该版本支持8087协处理器的浮点运算。

TURBOBCD.COM Turbo Pascal系统的BCD版本。该版本支持BCD实数格式。

CTURBO.COM Turbo Pascal系统的汉化版本。

CTURBO87.COM Turbo-87的汉化版本。

Graph.P 包含了对Graph.BIN文件中所使用的扩展图形功能和海龟作图子程序的外部说明。如果需要使用海龟作图或图形工具箱，则需要将它放在当前盘上，并在程序中用编译开关{\$I Graph.P}嵌入。

Graph.BIN 包含了扩展图形和海龟作图的机器语言例行子程序。如果需要使用扩展作图或海龟作图功能，则应将它放在当前盘上。

Read.me 包含了最新系统的勘误和建议。

在以上文件中，只有TURBO.COM文件必须放在当前工作盘上。整个Turbo Pascal系统只须37K字节空间。如果在编译时需要显示错误信息，则需要TURBO.MSG文件。TINST.COM和TINST.MSG文件只用于安装过程，Graph.P和Graph.BIN只在需要扩展图形或海龟作图功能时才使用，TURBO-87.COM、TURBOBCD.COM、CTURBO.COM和CTURBO87.COM只有在用相应的版本时才需要。

除了上述文件外，3.0版的Turbo Pascal系统盘上还包括如下文件：

ACCESS3.BOX 该文件包含了索引文件的建立、写入、读出等例程。

ART.PAS 随机作图示例。

CALC.PAS 电子表格程序。

CALC.HEP 电子表格帮助文件。

CALCDEMO.MCS 电子报表数据文件。

CALCMAIN.PAS 为编译电子报表程序而准备的主程序(当内存不够时使用它)。

CMDLIN.PAS 从命令行中取参数示例。

COLOR.PAS 彩色示例程序。

DOSFCALL.DOC 系统调用示例(从DOS取日期)。

DIRECT.PAS Turbo目录过程使用示例。

EXTERNAL.DOC 使用外部例程示例。

INTRPTCL.DOC 系统中断调用示例(从DOS取时间)。

LISTER.PAS 列出Pascal源程序的示例。

SOUND.PAS 声音示例。

TURTLE.PAS 海龟作图示例。

WINDOW.PAS 窗口示例。

以上文件中,有很多程序是可以执行的完整的程序,初学者可以打印出源程序,读懂并运行之,这对加快了解Turbo Pascal系统大有帮助。

§ 1.3 安装系统

在运行Turbo Pascal之前,可以用TINST程序来安装系统,使之适应所需环境。TINST安装程序可设置三方面的内容:

1. 设置屏幕类型。可以设置为黑白显示、彩色显示等类型的屏幕。
2. 定义编辑键。可以按照用户自己的习惯定义Turbo的各种编辑键。
3. 指定错误信息文件TURBO.MSG的磁盘路径名。为了减轻工作盘的存贮负担,可以将TURBO.MSG文件放在其它的盘上,通过TINST指明这个文件的磁盘路径名,以便由Turbo调用。

在操作系统提示符下,键入命令:

TINST ↵

来启动TINST程序, TINST首先显示主菜单:

TURBO Pascal installation menu.

Choose installation item from the following:

[S]creen type | [C]ommand installation | [M]sg file path | [Q]uit

Enter S, C, M or Q:

在主菜单下,可以键入S、C、M或Q中的任何一个命令。其中S为设置屏幕类型的命令,C为定义编辑键的命令,M为指定错误信息文件磁盘路径名的命令。Q为退出TINST程序,安装结束的命令。

下面分别介绍S、C和M三个命令的操作过程。

S——设置屏幕类型。键入S后, Tinst显示一个屏幕类型选择菜单:

Choose one of the following displays:

- 0) Default display mode
- 1) Monochrome display
- 2) Color display 80×25
- 3) Color display 40×25
- 4) b/w display 80×25

5) b/w display 40×25

Which display? (Enter no. or ^Q to exit).

你可根据显示屏类型键入0~5中某一个键, 其中0代表默认显示模式, 即保持进入Turbo之前的模式; 1为单色显示; 2为80×25彩色显示; 3为40×25彩色显示; 4为80×25黑白显示; 5为40×25黑白显示。也可以键入^Q, 表示不改变以前的设置。

选择了显示屏类型后, Tinst接着询问:

Does your screen blink when the text scrolls ? (Y/N)

(在文本滚动时屏幕是否闪烁?), 你可以根据在使用Turbo编辑器时, 屏幕滚动时是否闪烁来回答是(Y)或否(N)。

C——定义编辑键。

Turbo编辑器由一系列命令来操作, 它们分为“光标移动命令”、“插入删除命令”、“块处理命令”和“其它编辑命令”四部分, 其中的辅助命令是遵照Wordstar的“标准集”设置的。主命令是多数系统没有定义的, 但可以由Tinst来设置, 很容易定义以适合用户的习惯和键盘。IBMPC系统可以将箭头(↑ ↓ ← →)和功能键设置为主命令。

键入C后, Tinst显示:

CURSOR MOVEMENTS:

1: Character left <ESC> K →

这是提请你定义第一个编辑键——“左移一个字符”, 原来的设置为‘<ESC> K’, 即←键的控制序列。此时可以有几种回答方式:

(1) 按回车键, 表示不改变以前的设置。

(2) 键入一个控制键, 后跟回车。比如: 键入^L(ctrl键与L键同时按下), 则将^L定义为“左移一个字符”的控制键。

(3) 键入减号-, 表示取消该控制键。

定义完第一个控制键后, Tinst又让你定义第二个控制键。照此下去, 直至45个编辑键全部定义完为止。

注意: 不要重复定义编辑键, 否则Tinst会让你反复修改, 直至没有重复为止。

M——指定错误信息文件的磁盘路径名。键入M后, Tinst显示:

MSG file path and name (eg.A.: TURBO.MSG)

A: TURBO.MSG->

假设我们已将TURBO.MSG文件放在C盘的TURBO子目录下, 则可键入:

C: \TURBO\TURBO.MSG\

上述三方面的设置结束后, 键入Q, 则完成了Tinst的设置工作。

§ 1.4 编辑、编译和运行

这一节详细讲述如何使用Turbo系统, 包括如何启动, 如何进入编辑、编译和运行状态。在§ 1.5节将详细介绍各个编辑键的功能。

在使用Turbo系统之前, 最好先用COPY或DISKCOPY命令拷贝一个系统盘, 以备份盘作为工作盘, 而保存好原系统盘。

在DOS系统下, 插入Turbo系统盘, 键入命令:

TURBO \

即启动了Turbo系统。系统首先在屏幕上显示如下信息:

```
.....
TURBO Pascal system      Version 3.01A
                           PC-DOS
Copyright (C) 1983, 84, 85  BORLAND Inc.
.....
```

Default display mode

Include error messages (Y/N)?

这里显示了Turbo系统的版本号(3.01A)、运行环境(PC-DOS)以及各版本发行时间(分别为83, 84, 85)和公司名称(BORLAND Inc.), 还显示了屏幕设置模式(Default display mode), 这是由Tinst所设置的。

最后一行系统提问:“是否包括错误信息(Y/N)?”, 如果回答Y, 系统便将Turbo.msg文件读入内存, 并显示:

Loading Turbo.msg

如果回答N, 则不读入Turbo.msg文件。如果读入了Turbo.msg文件, 系统在编译源程序时, 若发现了源程序错, 则显示错误码和错误信息, 否则只显示错误码。对于有经验的程序员来说, 可以回答N, 这样可以节约大约1.5K字节的内存。

回答完这个问题后, 你便会看到Turbo的主菜单:

```
Logged drive:A
Active directory:\
Work file:
Main file:
Edit      Compile Run Save
Dir       Quit compiler Options
Text:      0 bytes
Free: 62024 bytes
>
```

从这个菜单你可以看出, 当前驱动器(Logged drive)是A, 当前目录(Active directory)为根目录(\)。这个菜单还列出了你现在所能使用的全部操作命令, 我们将这些命令及其功能和代表字母排列如下:

命 令	代表字母	功 能
Logged drive	L	选择当前驱动器
Active directory	A	选择当前目录
Work file	W	选择工作文件
Main file	M	选择主文件
Edit	E	编辑
Compile	C	编译
Run	R	运行
Save	S	保存工作文件
Dir	D	列目录
Quit	Q	退出Turbo系统
compiler Options	O	编译选择

现在可以执行其中的任意一条命令，而且只须键入相应的代表字母就行了。例如为了指定工作文件，只须键入W即可。细心的读者会发现，代表字母就是命令中的那个大写字母。在有些系统中，这个大写字母还是高亮度的。执行完任何一条命令（除Q外），系统又自动回到这个主菜单，等待执行下一条命令。

下面我们将详细介绍每条命令。

L——选择当前驱动器 L命令用来改变当前驱动器。按L键后，显示：

New drive: _

你可输入一个驱动器（A~P中任意一个）字母，后跟回车键。若不想改变当前驱动器，则只敲回车键即可。L命令完成盘的设置，即使不改变驱动器而只改变软盘，为了防止意外地写错盘，也应使用L命令。

执行完L命令后，菜单上的驱动器字母并不自动更新，如果希望菜单上显示新的当前驱动器号，可以敲一下空格键。

A——选择当前目录 A命令用来改变当前目录。按A键后，显示：

Active directory: _

你可以输入一个目录名，后跟一个回车键，比如：\ABC\，若不想改变当前目录，只敲回车键即可。

在进行了当前驱动器和当前目录的设置后，除了明确指出外，所有的文件（工作文件、主文件、.com文件、.chn文件等）均指当前驱动器的当前目录下的文件。

W——选择工作文件 W命令用来选择工作文件。以后的编辑(E)、编译(C)、运行(R)和存盘(S)等命令均是针对这个工作文件而言的。（若指定了主文件，则编译(C)和运行(R)命令针对主文件）

按W键后，系统询问：

Work file name: _

这时可以输入一个合法的文件名，文件名要求是由最多8个字符组成的名字，后跟句点“.”和不超过3个字符的后缀所组成。例如：

filename . typ

如果只输入文件名而不带句点和后缀，系统则自动在该文件名后加上句点和后缀pas。如果文件名后面跟了句点而没有带后缀，系统则不自动加上pas后缀。例如：

PROGRAM 变成PROGRAM.PAS

PROGRAM . 不变

PROGRAM.FIL 不变

注意：应避免使用扩展名.BAK，.CHN和.COM作为后缀，因为这是一些专用后缀，.BAK是备份文件的后缀，.CHN是链文件的后缀，.COM是可执行文件的后缀。

指定了工作文件名后，如果盘上有该文件，系统就将它读入内存，如果没有这个文件，系统便显示：

New File

并清除编辑所用的内存。

如果在选择工作文件之前，已经编了一个工作文件，假设为OLD.PAS，而且这个文件未曾存盘，系统将提示你是否要存盘：

Workfile OLD.PAS not saved. Save (Y/N)? _

请注意，一旦调入新的工作文件，则会覆盖掉以前的工作文件所占用的内存，如果希望保存以前的文件，则应回答Y，否则回答N。

M——选择主文件 M命令用来定义一个主文件。若程序太大，不宜用一个文件时，就使用几个文件，而其中有一个为主文件，通过编译开关\$I指明嵌入其它文件(详见§8.1节)。在编译这种程序时，我们可以用M命令来指明这个主文件，用户可以定义其它文件为工作文件，这样，可以保持主文件不变而编辑其它的嵌入文件。

当执行编译命令时，如果工作文件与主文件不同，系统则首先自动保存工作文件，装入主文件，在编译期间，如果发现错误，系统就将那个有错误的文件(不论是主文件还是嵌入文件)自动变为工作文件，供用户对它进行修改，改正错误后若再进行编译，被修改的文件自动存盘，系统重新装入主文件进行编译。

主文件名和工作文件名的格式要求相同。

E——编辑 E命令用来调用Turbo的内部编辑器，用它来编辑当前的工作文件。如果之前没有指定工作文件，系统会提示用户指定工作文件名，主菜单消失后，编辑器便开始工作。在编辑状态下，你可以使用所提供的编辑键来编写、修改你的程序，直至你用^KD结束编辑返回到主菜单为止。

C——编译 C命令用来启动编译器。若指定了主文件，则编译主文件，否则编译工作文件。如果指定了主文件而且已编辑了工作文件，则在装入和编译主文件之前，系统会询问是否保存工作文件。在编译过程中，敲任意键即可中断编译。

在编译时，系统若发现源程序有错，则会将有错的文件作为工作文件，提示用户按<ESC>键后，自动进入编辑状态，并将光标定位在出现错误的那个位置，修改并退出编辑状态后，还可重新编译这个程序。

编译可以在内存中产生可执行文件，也可以形成.COM文件或.CHN文件并存放在磁盘中，这要根据编译选择(参见O命令)而定。内存方式是系统的默认方式。

R——运行 R命令用来运行一个程序。注意，R命令只在内存方式(参见O命令)才能运行程序。

键入R后，若在这之前没有指定工作文件也没有指定主文件，系统则首先询问工作文件名。

若内存中的程序在执行R之前未经编译，系统则先编译这个程序，然后再运行。

在运行过程中，若发生运行错误，则显示错误信息，提示输入<ESC>键后，以出错文件为工作文件，自动进入编辑状态，供用户修改。退出编辑状态后，可以重新运行这个程序。

S——存盘 S命令用来将当前工作文件存贮到磁盘上。磁盘中该文件的旧版本将改名为.BAK文件。

D——列目录 D命令用来显示磁盘文件目录表以及剩余磁盘空间等信息。按D键后，系统询问：

Dir mask: _

用户可以输入一个目录掩码后跟回车键，或仅仅只输入一个回车键。掩码可以由驱动器号、磁盘目录名、文件名和后缀四部分组成。如果不给定驱动器号，则显示当前驱动器的目录表，如果不给定目录名，则显示当前目录的目录表，文件名和后缀中还可以使用通配符*和?。

如果只按回车键，则显示当前驱动器的当前目录的目录表。

Q——退出Turbo系统 Q命令用来退出Turbo系统。如果已编辑一个文件但未存盘，系统

在退出前将询问是否保存该文件。

O——编译选择 O命令用来选择编译方式和修改一些编译默认值。按O键后，屏幕上出现如下菜单：

```
compile-> Memory
          Com-file
          cHn-file
command line Parameters,
Find run-time error Quit
>
```

用↑、↓键将箭头→移至要选的行上或键入M、C、H可以选择三种编译方式中的一种，即指明将编译的目标代码置于何处。Compile→所指的行便是当前所选择的编译方式，其中Memory为内存方式，Com-file为COM文件方式，cHn-file为链文件方式。

系统的默认方式为M，即内存方式，在这种方式下，编译后的目标代码放在内存中，可以用R命令立即运行该程序。目标代码不存到磁盘上，如果停电或退出系统，该目标码则被冲掉。

输入C则为COM命令文件方式，这时箭头→移到Com-file这一行，编译后的目标代码将写至与工作文件名（或主文件名）相同且后缀为COM的磁盘文件中，可以在操作系统命令状态下键入该文件名来运行这个程序。

输入H则为CHN（链）文件方式，这时箭头→移至cHn-file这一行，编译后的目标代码仅包含程序代码，而不包含Pascal库，目标代码被写至与工作文件名（或主文件名）相同且后缀为CHN的磁盘文件中（参见§8.3）。

键入P命令，定义在内存方式下的命令行参数。例如，我们编写一个复制程序COPY2，它在操作系统下的调用格式为：

```
COPY2 File1 File2
```

其中命令行参数File1、File2定义了拷贝的输入文件和输出文件的名称。若在内存方式下运行，怎样才能输入File1、File2这两个文件名呢？我们可以通过键入P命令，然后输入File1、File2，以此作为命令行参数。

键入F命令即查找运行时错误。在内存方式下运行时，一旦出现错误，系统会自动找到出错位置。但在编译成COM和CHN文件再运行程序时，若发现错误，则无法自动进入编辑状态，从而无法找到出错位置，此时系统显示如下信息：

```
Run time error 01 PC=1B56
program aborted
```

为了找出出错位置：你可用纸记下出错时的PC计数器值1B56，然后进入Turbo系统，调入源程序，按O键进入编译选择子菜单，然后再按F键，系统便提示：

```
Enter PC;
```

这时你再键入1B56，系统就会指出出错位置。

当键入C键后，屏幕上出现如下菜单：

```
minimum cOde segment size: 0000 (max 0D28 paragraphs)
minimum Data segment size: 0000 (max 0FDC paragraphs)
mInimum free dynamic memory: 0400 paragraphs
mAximum free dynamic memory: A000 paragraphs
```


按O键,则设置最小代码段尺寸,按D键,则设置最小数据段尺寸,按I键,则设置堆栈的最小内存,按A键,则设置堆栈的最大内存。最小代码段尺寸和最小数据段尺寸如何设置,我们将在§8.3节中介绍Chain和Execute过程时举例说明。

按Q键则退出编译选择子菜单,返回到Turbo主菜单。

至此,我们介绍完了在Turbo主菜单上所提供的的所有命令。

§ 1.5 Turbo 编辑器

Turbo的内部编辑器是专为编制源程序文本而设计的全屏幕编辑器。如果你熟悉Micropro公司的Wordstar,那么只要稍加指点就能使用这个编辑器。除了只有一些细小的差别和扩充外,Turbo编辑器的编辑命令几乎与Wordstar的一模一样。除了wordstar的命令之外,在IBM PC系统上还设置了箭头键和功能键,你还可以按照§1.3中介绍的方法设置一些自己的编辑命令。设置后,仍然可以使用Wordstar的命令。

Turbo编辑器的使用非常简单,在定义了工作文件之后,敲E键,主菜单消失后编辑器即开始工作。若工作文件在当前驱动器上,系统则将它装入内存并显示其第一页。如果是新文件,系统则清屏,只剩下屏幕顶部的状态行。

按^KD键(即Ctrl-K-D,同时按<Ctrl>与K键,然后按D键)即可退出编辑器并返回到Turbo主菜单(后面还会提到^KD键)。

用键盘输入源程序文本同使用打字机很相似,结束一行时,按回车键(或者称<RETURN>键、<CR>键或<ENTER>键等,在书中常用符号↵代替)。当输入的各行填满了整个屏幕时,顶行会卷离屏幕,但这一行并没有丢失,可以用后面将介绍的编辑命令在文本中前后翻页。

先了解一下屏幕顶上状态行的意义。屏幕顶上的那行称为状态行,它包含如下信息:

Line n Col n Insert Indent X: Filename.TYP

其中:

Line n——显示光标所在行的行号(从文件开始处算起)。

Col n——显示光标所在位置的列号(从该行左边算起)。

Insert——插入状态。在这个状态下,从键盘所键入的字符将被插入在光标位置上,光标右边原有的文字同时往右移动。使用插入开关命令(默认键为^V)将使Insert变为Overwrite,即重写状态。在这种状态下,从键盘所键入的字符将重写至光标位置,不作插入。

Indent——指明处于自动缩进状态。可以用自动缩进开关命令(默认为^QI)退出缩进状态。在缩进状态下,在产生新的一行时,光标自动与上一行的第一个非空字符对齐,这可以帮助编写有一定缩进格式的程序。

X: Filename.typ——显示被编辑的文件名,包括驱动器号、文件名称以及后缀。

下面我们来逐一介绍Turbo编辑器的编辑命令。

如前所述,使用编辑器与使用打字机很相似。但由于它是计算机化的文本编辑器,所以它还提供了很多其它的编辑功能,有了这些功能,处理文本特别是编写程序比起在纸上打印要容易得多。

Turbo编辑器共接受45个编辑命令,其中有两个是辅助命令,用这些命令可移动光标、翻页、查找和替换文本字符串等。这些命令可以分为下述四组:

光标移动命令 插入与删除命令