

Turbo C 2.0

实用高级编程技巧

王军政 编



北京科海培训中心

Turbo C 2.0 实用高级编程技巧

王军政 编

JS/01/28

北京科海培训中心

封面设计：夏 峰

订购单位：北京海淀路82号科海培训中心

邮 码：100080

电 话：2562449 2562954

乘 车：320、332、302路黄庄站

下车海淀剧院北马路往里80米

地 点：海淀文化馆平房资料销售部

前 言

C 语言以其强大的功能成为计算机语言中的佼佼者，而 Turbo C 则以高效的集成开发环境更赢得了广大用户的喜爱。

近几年国内出版了不少关于 Turbo C 语言及其应用的专著，但对于使用 Turbo C 编写显示、打印汉字程序的方法、中文下拉式菜单的设计以及 Turbo C 与 FOXBASE 和汇编语言的接口等技术作全面、系统、详细介绍的书籍几乎没有，可是这些技术非常实用而且普遍。为此，本书作者将其在实际使用 Turbo C 编程所积累的一些经验和体会汇集成册，奉献给广大 Turbo C 爱好者。

本书分为两大部分共十三章。第一部分介绍 Turbo C 2.0 语言，主要是为初学者编写的，但其中一些内容，如连续使用字符输入函数时存在的问题及解决途径、键盘的动态扫描、BIOS、DOS 功能调用方法、程序的调试等内容对熟悉 Turbo C 的读者也是非常有用的。第二部分介绍 Turbo C 语言的应用技巧。内容包括：西文操作系统下读取 16×16 点阵字模、 24×24 点阵字模按水平或垂直方向屏幕显示或放大显示不同字体、不同颜色的汉字技术；读取 FOXBASE 数据库或从 FOXBASE 以传递参数的办法调用 Turbo C 程序技术；高分辨率屏幕图形的打印机输出和打印机驱动程序编写技术；中文下拉式菜单的设计技术；调用汇编子程序和行间嵌入汇编语句的 Turbo C 程序编写及编译、连接技术。

本书列举大量实例，全部例程均用中文注释。每个例程都是作者亲自编写并已调试通过，可直接被读者使用。

由于作者水平所限，经验不足，加之时间仓促，书中难免有不少错误和疏漏，诚恳接受广大读者和有关专家的批评指正。

本书在出版过程中，得到了科海培训中心华根娣、夏非彼两位老师的大力协助，在此向她们表示衷心的感谢！

编 者

一九九二年五月

目 录

第一部分 Turbo C 2.0 语言

第一章 Turbo C 2.0 集成开发环境	(1)
1.1 C 语言概述	(1)
1.1.1 C 语言的产生与发展	(1)
1.1.2 C 语言的特点	(1)
1.2 Turbo C 2.0 概述	(2)
1.2.1 Turbo C 的产生与发展	(2)
1.2.2 Turbo C 2.0 基本配置要求	(2)
1.2.3 Turbo C 2.0 软盘内容简介	(2)
1.3 学习本书应具有的软件环境	(3)
1.4 Turbo C 2.0 的安装和启动	(4)
1.5 Turbo C 2.0 集成开发环境的使用	(4)
1.5.1 主菜单	(5)
1.5.2 Turbo C 的配置文件	(11)
第二章 数据类型、变量和运算符	(12)
2.1 Turbo C 程序的一般组成部分	(12)
2.2 数据类型	(14)
2.2.1 整型 (int)	(15)
2.2.2 浮点型 (float)	(15)
2.2.3 字符型 (char)	(16)
2.2.4 指针型 (*)	(17)
2.2.5 无值型 (void)	(17)
2.3 关键字和标识符	(17)
2.3.1 关键字	(17)
2.3.2 标识符	(18)
2.4 变量	(18)
2.4.1 变量说明	(18)
2.4.2 变量种类	(18)
2.4.3 变量存储类型	(20)
2.4.4 数组变量	(21)
2.4.5 变量的初始化和赋使	(22)
2.5 运算符	(28)
2.5.1 算术运算符	(28)

2.5.2	关系运算符和逻辑运算符	(30)
2.5.3	按位运算符	(31)
2.5.4	Turbo C 的特殊运算符	(32)
2.5.5	Turbo C 运算符的优先级	(33)
第三章	输入输出函数	(35)
3.1	标准输入输出函数	(35)
3.1.1	格式化输入输出函数	(35)
3.1.2	非格式化输入输出函数	(39)
3.2	文件的输入输出函数	(42)
3.2.1	标准文件函数	(43)
3.2.2	非标准文件函数	(49)
第四章	控制流程语句	(51)
4.1	条件语句	(51)
4.2	循环语句	(52)
4.2.1	for 循环	(52)
4.2.2	while 循环	(53)
4.2.3	do while 循环	(55)
4.3	开关语句	(55)
4.4	break、continue 和 goto 语句	(57)
4.4.1	break 语句	(57)
4.4.2	continue 语句	(57)
4.4.3	goto 语句	(58)
第五章	结构、联合和枚举	(60)
5.1	结构 (struct)	(60)
5.1.1	结构说明和结构变量定义	(60)
5.1.2	结构变量的使用	(61)
5.1.3	结构数组和结构指针	(63)
5.1.4	结构的复杂形式	(65)
5.2	联合 (union)	(67)
5.2.1	联合说明和联合变量定义	(67)
5.2.2	结构和联合的区别	(68)
5.3	枚举	(69)
5.4	类型说明	(70)
5.5	预处理指令	(71)
第六章	函数	(74)
6.1	函数的说明和定义	(74)
6.1.1	函数说明	(74)
6.1.2	函数定义	(75)
6.2	函数的调用	(75)

6.2.1 函数的简单调用	(75)
6.2.2 函数的参数传递	(76)
6.2.3 函数的递归调用	(81)
6.3 函数的作用范围	(82)
第七章 字符屏幕和图形函数	(83)
7.1 字符屏幕函数	(83)
7.1.1 文本窗口的定义	(83)
7.1.2 文本窗口颜色的设置	(83)
7.1.3 窗口内文本的输入输出函数	(85)
7.1.4 有关屏幕操作的函数	(86)
7.2 图形函数	(88)
7.2.1 图形模式的初始化	(88)
7.2.2 独立图形运行程序的建立	(90)
7.2.3 屏幕颜色的设置和清屏函数	(91)
7.2.4 基本图形函数	(93)
7.2.5 封闭图形的填充	(96)
7.2.6 有关图形窗口和图形屏幕操作函数	(100)
7.2.7 图形模式下的文本输出	(103)
第八章 Turbo C 实用编程	(107)
8.1 汉字操作系统下含有汉字输入输出的程序编制	(107)
8.2 Turbo C 提供的 BIOS、DOS 系统的调用函数	(111)
8.2.1 关于键盘操作的函数 bioskey()	(111)
8.2.2 关于打印机操作的函数 biosprint()	(114)
8.2.3 关于 DOS 软中断功能调用的函数 intdos()	(117)
8.2.4 关于 BIOS、DOS 软中断调用的函数 int86()	(118)
8.2.5 其它一些系统调用函数	(118)
8.3 有关字符串函数、数字字符串与数值的转换函数	(120)
8.3.1 有关字符串的函数	(120)
8.3.2 数字字符串与数值的转换函数	(122)
8.4 Turbo C 的动态内存分配、过程控制和数学运算函数	(123)
8.4.1 动态内存分配函数	(123)
8.4.2 过程控制函数	(124)
8.4.3 数学运算函数	(126)
8.5 Turbo C 集成开发环境下程序的调试	(126)
8.5.1 编译时的常见错误	(127)
8.5.2 连接时的常见错误	(127)
8.5.3 运行时的常见错误	(127)
8.6 Turbo C 的命令行编译	(128)

第二部分 Turbo C 2.0 应用技术专题

第九章 西文操作系统下显示汉字技术	(130)
9.1 在西文状态下显示 16×16 点阵汉字	(130)
9.1.1 16×16 点阵汉字字模存储格式	(130)
9.1.2 西文状态下显示 16×16 点阵汉字的实现	(130)
9.2 在西文状态下显示 24×24 点阵汉字	(136)
9.2.1 24×24 点阵汉字字模存储格式	(136)
9.2.2 西文状态下显示 24×24 点阵汉字的实现	(137)
9.3 在西文状态下显示按任意倍数放大不同字体的 24X24 点阵汉字	(141)
第十章 与 FOXBASE(DBASE)接口技术	(147)
10.1 Turbo C 直接读取 FOXBASE 数据库中的数据	(147)
10.1.1 FOXBASE 数据库的结构	(147)
10.1.2 Turbo C 读取数据库中数据的实现	(149)
10.2 FOXBASE 给 Turbo C 传递参数	(152)
第十一章 Turbo C 的高级打印技术	(155)
11.1 利用打印机驱动程序放大打印汉字	(155)
11.2 VGA 高分辨率(640×480)屏幕图形的打印机输出	(157)
11.2.1 M1724 打印机的控制命令介绍	(157)
11.2.2 VGA 高分辨率屏幕图形的打印机输出程序	(158)
11.3 自编打印机驱动程序	(164)
第十二章 菜单设计技术	(173)
12.1 西文下拉式菜单的设计	(173)
12.2 中文窗口式菜单的设计	(177)
12.3 FOXBASE 和 Turbo C 程序交替使用时菜单的设计	(181)
12.4 中文下拉式菜单的设计	(186)
第十三章 与汇编语言的接口技术	(193)
13.1 Turbo C 调用汇编子程序	(193)
13.1.1 Turbo C 与汇编语言的接口方法	(193)
13.1.2 自动产生汇编语言的框架程序	(196)
13.1.3 编译、连接、运行接口程序	(199)
13.2 Turbo C 行间嵌入汇编	(200)
附录 ASCII 字符代码表	(203)

第一部分 Turbo C 2.0 语言

这一部分详细地讲述 Turbo C 2.0 语言。主要包括：集成开发环境的使用、变量类型、操作运算、输入输出函数、控制流程语句以及字符和图形屏幕函数等内容。

第一章 Turbo C 2.0 集成开发环境

本章首先介绍 C 和 Turbo C 的特点以及 Turbo C 2.0 的软盘内容、安装方法，最后介绍 Turbo C 的集成开发环境。通过本章的学习，读者可以全面系统地学习 Turbo C 集成开发环境的使用方法，这也是学习 Turbo C 必不可少的。

1.1 C 语言概述

1.1.1 C 语言的产生与发展

C 语言是 1972 年由美国的 Dennis Ritchie 设计发明的，并首次在 UNIX 操作系统的 DEC PDP-11 计算机上使用。它由早期的编程语言 BCPL(Basic Combind Programming Language)发展演变而来。在 1970 年，AT&T 贝尔实验室的 Ken Thompson 根据 BCPL 语言设计出较先进的并取名为 B 的语言，最后导致了 C 语言的问世。

随着微型计算机的日益普及，出现了许多 C 语言版本。由于没有统一的标准，使得这些 C 语言之间出现了一些不一致的地方。为了改变这种情况，美国国家标准研究所(ANSI)为 C 语言制定了一套 ANSI 标准，成为现行的 C 语言标准。

1.1.2 C 语言的特点

C 语言发展如此迅速，而且成为最受欢迎的语言之一，主要因为它具有强大的功能。许多著名的系统软件，如 DBASE III PLUS、DBASE IV 都是由 C 语言编写的。用 C 语言加上一些汇编语言子程序，就更能显示 C 语言的优势了，象 PC-DOS、WORDSTAR 等就是用这种方法编写的。归纳起来 C 语言具有下列特点：

1. C 是中级语言

它把高级语言的基本结构和语句与低级语言的实用性结合起来。C 语言可以象汇编语言一样对位、字节和地址进行操作，而这三者是计算机最基本的工作单元。

2. C 是结构式语言

结构式语言的显著特点是代码及数据的分隔化，即程序的各个部分除了必要的信息交流外彼此独立。这种结构化方式可使程序层次清晰，便于使用、维护以及调试。C 语言是

以函数形式提供给用户的，这些函数可被用户方便的调用，并具有多种循环、条件语句控制程序流向，从而使程序完全结构化。

3. C 语言功能齐全

C 语言具有各种各样的数据类型，并引入了指针概念，可使程序效率更高。另外 C 语言也具有强大的图形功能，支持多种显示器和驱动器。而且计算功能、逻辑判断功能也比较强大，可以实现决策目的。

4. C 语言适用范围大

C 语言还有一个突出的优点就是适合于多种操作系统，如 DOS、UNIX，也适用于多种机型。

1.2 Turbo C 概述

1.2.1 Turbo C 的产生与发展

Turbo C 是美国 Borland 公司的产品，Borland 公司是一家专门从事软件开发、研制的大公司。该公司相继推出了一套 Turbo 系列软件，如 Turbo BASIC、Turbo Pascal、Turbo Prolog，这些软件很受用户欢迎。该公司在 1987 年首次推出了 Turbo C 1.0 产品，其中使用了全然一新的集成开发环境，即使用了一系列下拉式菜单，将文本编辑、程序编译、连接以及程序运行一体化，大大方便了程序的开发。1988 年，Borland 公司又推出 Turbo C 1.5 版本，增加了图形库和文本窗口函数库等，而 Turbo C 2.0 则是该公司 1989 年出版的。Turbo C 2.0 在原来集成开发环境的基础上增加了查错功能，并可以在 Tiny 模式下直接生成.COM(数据、代码、堆栈处在同一 64K 内存中)文件。还可对数学协处理器(支持 8087/80287/80387 等)进行仿真。

Borland 公司后来又推出了面向对象的程序软件包 Turbo C++，它继承发展了 Turbo C 2.0 的集成开发环境，并包含了面向对象的基本思想和设计方法。

1991 年为了适用 Microsoft 公司的 Windows 3.0 版本，Borland 公司又将 Turbo C++作了更新，即 Turbo C 的新一代产品 Borland C++ 2.0 也已经问世了。

1.2.2 Turbo C 2.0 基本配置要求

Turbo C 2.0 可运行于 IBM-PC 系列微机，包括 XT、AT 及 IBM 兼容机。此时要求 DOS 2.0 或更高版本支持，并至少需要 448K 的 RAM，可在任何彩、单色 80 列监视器上运行。支持数学协处理器芯片，也可进行浮点仿真，这将加快程序的执行。

1.2.3 Turbo C 2.0 软盘内容简介

Turbo C 2.0 有六张低密软盘(或两张高密软盘)。下面对 Turbo C 2.0 的主要文件作一简单介绍：

INSTALL.EXE	安装程序文件
TC.EXE	集成编译器
TCINST.EXE	集成开发环境的配置设置程序
TCHelp.TCH	帮助文件

THELP.COM	读取TCHELP.TCH的驻留程序
README	关于Turbo C的信息文件
TCCONFIG.EXE	配置文件转换程序
MAKE.EXE	项目管理工具
TCC.EXE	命令行编译器
ILINK.EXE	Turbo系列连接器
TLIB.EXE	Turbo系列库管理工具
C0?.OBJ	不同模式启动代码
C?.LIB	不同模式运行库
GRAPHICS.LIB	图形库
EMU.LIB	8087仿真库
FP87.LIB	8087库
*.H	Turbo C头文件
*.BGI	不同显示器图形驱动程序
*.C	Turbo C例行程序(源文件)

其中: 上面的?分别为:

T	Tiny(微型模式)
S	Small(小模式)
C	Compact(紧凑模式)
M	Medium(中型模式)
L	Large(大模式)
H	Huge(巨大模式)

1.3 学习本书应具有的软硬件环境

本书以 Turbo C 2.0 为蓝本, 因此必须有一套齐全的 Turbo C 2.0 版本。除此之外, 要很顺利的使用本书中的例程, 特别是第二部分的应用专题例行程序, 应具有以下软、硬件配置。

硬件:	80286(或兼容机)主机
	VGA(或TVGA)分辨率(640×480)显示器
	M1724打印机
软件:	DOS 3.0以上版本操作系统
	UCDOS 2.0汉字操作系统
	FOXBASE 2.0数据库管理语言
	MASM 4.0以上宏汇编编译程序

另外, 本书介绍的函数和语句许多是 C 语言标准, 即其它 C 语言中也有, 但作者未对此进行区分而叙述成 Turbo C 的函数和语句, 特此说明。

1.4 Turbo C 2.0 的安装和启动

Turbo C 2.0 的安装非常简单, 只要将 1 盘插入 A 驱动器中, 在 DOS 的 "A>" 下键入:

A>INSTALL

即可，此时屏幕上显示三种选择：

1. 在硬盘上创建一个新目录来安装整个 Turbo C 2.0 系统。
2. 对 Turbo C 1.5 更新版本。

这样的安装将保留原来对选择项、颜色和编辑功能键的设置。

3. 为只有两个软盘而无硬盘的系统安装 Turbo C 2.0。

· 这里假定按第一种选择进行安装，只要在安装过程中按对盘号的提示，顺序插入各个软盘，就可以顺利地进行安装，安装完毕将在 C 盘根目录下建立一个 TC 子目录，TC 下还建立了两个子目录 LIB 和 INCLUDE，LIB 子目录中存放库文件，INCLUDE 子目录中存放所有头部文件。

运行 Turbo C 时, 只要在 TC 子目录下键入 TC 并回车即可进入 Turbo C 集成开发环境。

1.5 Turbo C 2.0 集成开发环境的使用

进入 Turbo C 集成开发环境中后，屏幕上显示：

```
File Edit Run Compile Project Options Debug Break/watch
Edit
Line 1 Col 1 Insert Indent Tab File Unindent C:\NONAME.C

Message

F1 -Help F5 -Zoom F6 -Switch F7 -Trace F8 -Step F9 -Make F10 -Menu
```

其中顶上 一行为 Turbo C 主菜单，中间窗口为编辑区，接下来是信息窗口，最底下一行为参考行。这四个窗口构成了 Turbo C 的主屏幕，以后的编程、编译、调试以及运

行都将在这个主屏幕中进行。下面详细介绍主菜单的内容。

1.5.1 主菜单

主菜单在 Turbo C 主屏幕顶上一行, 显示下列内容:

File Edit Run Compile Project Options Debug Break / watch

除 Edit 外, 其它各项均有子菜单, 只要用 Alt 加上某项中第一个字母(即大写字母), 就可进入该项的子菜单中。

一、File(文件)菜单

按 Alt+F 可进入 File 菜单, 该菜单包括以下内容:

• Load(加载)

装入一个文件, 可用类似 DOS 的通配符(如 *.C)来进行列表选择。也可装入其它扩展名的文件, 只要给出文件名(或只给路径)即可。该项的热键为 F3, 即只要在主菜单中按 F3 就可进入该项, 而不需要先进入 File 菜单再选此项。

• Pick(选择)

将最近装入编辑窗口的 8 个文件列成一个表让用户选择, 选择后将该程序装入编辑区, 并将光标置在上次修改过的地方。其热键为 Alt+F3。

• New(新文件)

说明文件是新的, 缺省文件名为 NONAME.C, 存盘时可改名。

• Save(存盘)

将编辑区中的文件存盘, 若文件名是 NONAME.C 时, 将询问是否更改文件名, 其热键为 F2。

• Write to(存盘)

可由用户给出文件名将编辑区中的文件存盘, 若该文件已存在, 则询问是否覆盖。

• Directory(目录)

显示目录及目录中的文件, 并可由用户选择。

• Change dir(改变目录)

显示当前目录, 用户可以改变显示的目录。

• Os shell(暂时退出)

暂时退出 Turbo C 到 DOS 提示符下, 此时可以运行 DOS 命令, 若想回到 Turbo C 中, 只要在 DOS 状态下键入 EXIT 即可。

• Quit(退出)

退出 Turbo C, 返回到 DOS 操作系统中, 其热键为 Alt+X。

注明:

●以上各项可用光标键移动色棒进行选择, 回车则执行。也可用每一项的第一个大写字母直接选择。若要退到主菜单或从它的下一级菜单列表框退回均可用 Esc 键, Turbo C 所有菜单均采用这种方法进行操作, 以下不再说明。

二、Edit(编辑)菜单

按 Alt+E 可进入编辑菜单, 若再回车, 则光标出现在编辑窗口, 此时用户可以进行文本编辑。

编辑方法基本与 Wordstar 相同, 可用 F1 键获得有关编辑方法的帮助信息。

与编辑有关的功能键如下:

F1	获得Turbo C编辑命令的帮助信息
F5	扩大编辑窗口到整个屏幕
F6	在编辑窗口与信息窗口之间进行切换
F10	从编辑窗口转到主菜单

编辑命令简介:

Page Up	向前翻页
Page Dn	向后翻页
Home	将光标移到所在行开始
End	将光标移到所在行结尾
Ctrl+Y	删除光标所在的一行
Ctrl+T	删除光标所在处的一个词
Ctrl+KB	设置块开始
Ctrl+KK	设置块结尾
Ctrl+KV	块移动
Ctrl+KC	块拷贝
Ctrl+KY	块删除
Ctrl+KR	读文件
Ctrl+KW	存文件
Ctrl+KP	块文件打印
Ctrl+F1	如果光标所在处为Turbo C库函数, 则获得有关该函数的帮助信息
Ctrl+Q[	查找Turbo C双界符的后匹配符
Ctrl+Q]	查找Turbo C双界符的前匹配符

注明:

● Turbo C 的双界符包括以下几种符号。

花括号	{ 和 }
尖括号	< 和 >
圆括号	(和)
方括号	[和]
注释符	/ * 和 * /
双引号	"
单引号	'

● Turbo C 在编辑文件时还有一种功能, 就是能够自动缩进。所谓自动缩进是指在键回车之后, 光标定位和上一个非空字符对齐。在编辑窗口中, Ctrl+OL 为自动缩进开关的控制键。

二、Run(运行)菜单

按 Alt+R 可进入 Run 菜单, 该菜单有以下各项:

- Run(运行程序)

运行由 Project / Project name 项指定的文件名或当前编辑区的文件。如果对上次编译后的源代码未做过修改, 则直接运行到下一个断点(没有断点则运行到结束)。否则先进行编译、连接后才运行, 其热键为 Ctrl+F9。

- Program reset(程序重启)

中止当前的调试, 释放分给程序的空间, 其热键为 Ctrl+F2。

- Go to cursor(运行到光标处)

调试程序时使用, 选择该项可使程序运行到光标所在行。光标所在行必须为一条可执行语句, 否则提示错误。其热键为 F4。

- Trace into(跟踪进入)

在执行一条调用其它用户定义的子函数时, 若用 Trace into 项, 则执行长条将跟踪到该子函数内部去执行, 其热键为 F7。

- Step over(单步执行)

执行当前函数的下一条语句, 即是用户函数调用, 执行长条也不会跟踪进函数内部, 其热键为 F8。

- User screen(用户屏幕)

显示程序运行时在屏幕上显示的结果。其热键为 Alt+F5。

四、Compile(编译)菜单

按 Alt+C 可进入 Compile 菜单, 该菜单有以下几个内容:

- Compile to OBJ(编译生成目标码)

将一个.C 源文件编译生成.OBJ 目标文件, 同时显示生成的文件名。其热键为 Alt+F9。

- Make EXE file(生成执行文件)

此命令生成一个.EXE 的文件, 并显示生成的.EXE 文件名。其中.EXE 文件名是下面几项之一。

1. 由 Project / Project name 说明的项目文件名。
2. 若没有项目文件名, 则由 Primary C file 说明的源文件。
3. 若以上两项都没有文件名, 则为当前窗口的文件名。

- Link EXE file(连接生成执行文件)

把当前.OBJ 的文件及库文件连接在一起生成.EXE 文件。

- Build all(建立所有文件)

重新编译项目里的所有文件, 并进行装配生成.EXE 文件。该命令不作过时检查(上面的几条命令要作过时检查, 即如果目前项目里源文件的日期和时间与目标文件相同或更早, 则拒绝对源文件进行编译)。

- Primary C file(主 C 文件)

当在该项中指定了主文件名后, 在以后的编译中, 如没有项目文件名则编译此项中指定的主 C 文件, 如果编译中有错误, 则将此文件调入编辑窗口, 不管目前窗口中是否是主 C 文件。

- Get info(获得信息)

该命令可获得有关当前路径、源文件名、源文件字节大小、编译中的错误数目、可用空间等信息。

五、Project(项目)菜单

按 Alt+P 可进入 Project 菜单，该菜单包括以下内容：

- Project name(项目名)

项目名具有 .PRJ 的扩展名，其中包括将要编译、连接的文件名。例如有一个程序由 file1.c、file2.c、file3.c 组成，要将这 3 个文件编译装配成一个 file.exe 的执行文件。可以先建立一个 file.prj 的项目文件，其内容如下：

```
file1.c
file2.c
file3.c
```

此时将 file.prj 放入 Project name 项中，以后进行编译时将自动对项目文件中规定的三个源文件分别进行编译。然后连接成 file.exe 的文件。

如果其中有些文件已经编译成 .OBJ 文件，而又没有修改过，可直接写上 .OBJ 扩展名。此时将不再编译而只进行连接。

例如：

```
file1.obj
file2.c
file3.c
```

将不对 file1.c 进行编译，而直接连接。

注明：

●当项目文件中的每个文件无扩展名时，均按源文件对待，另外，其中的文件也可以是库文件，但必须写上扩展名 .lib。

- Break make on(中止编译)

由用户选择是否有 Warning(警告)、Errors(错误)、Fatal Errors(致命错误)时或 Link(连接)之前退出 Make 编译。

- Auto dependencies(自动依赖)

当开关置为 on，编译时将检查源文件与对应的 .OBJ 文件日期和时间，否则不进行检查。

- Clear project(清除项目文件)

清除 Project / Project name 中的项目文件名。

- Remove messages(删除信息)

把错误信息从信息窗口中清除掉。

六、Options(选择)菜单

按 Alt+O 可进入 Options 菜单，该菜单对初学者来说要谨慎使用。

- Compiler(编译器)

本项选择又有许多子菜单，可以让用户选择硬件配置、存储模型、调试技术、代码优

化、对话信息控制和宏定义。这些子菜单如下:

Model

共有 Tiny,small,medium,compact,large,huge,model 七种不同内存模式可由用户选择。

Define

打开一个宏定义框,用户可输入宏定义。多重定义可用分号,赋值可用等号。

Code generation

它又有许多任选项,这些任选项告诉编译器产生什么样的目标代码。

Calling convention

可选择C或Pascal方式传递参数。

Instruction set

可选择8088/8086或80186/80286指令系列。

Floating point

可选择仿真浮点、数学协处理器浮点或无浮点运算。

Default char type

规定char的类型。

Alignment

规定地址对准原则。

Merge duplicate strings

作优化用,将重复的字符串合并在一起。

Standard stack frame

是否产生一个标准的栈结构。

Test stack overflow

是否产生一段程序运行时堆栈溢出的代码。

Line number

是否在.OBJ文件中放进行号以供调试时用。

OBJ debug information

是否在.OBJ文件中产生调试信息。

Optimization

Optimize for

选择是对程序小型化还是对程序速度进行优化处理。

Use register variable

用来选择是否允许使用寄存器变量。

Register optimization

尽可能的使用寄存器变量以减少过多的取数操作。

Jump optimization

通过去除多余的跳转和调整循环与开关语句的办法,压缩代码。

Source

Identifier length

说明标识符有效字符的个数,默认为32个。

Nested comments

是否允许嵌套注释。

ANSI keywords only

是只允许ANSI关键字还是也允许Turbo C关键字。

Error

Error stop after

多少个错误时停止编译,默认为25个。

Warning stop after

多少个警告错误时停止编译,默认为100个。

Display warning

此开关为on时将显示下列警告错误:

Portability warning

移植性警告错误。

ANSI Violations

侵犯了ANSI关键字的警告错误。

Common error

常见的警告错误。

Less common error

少见的警告错误。

Names

用于改变段(segment)、组(group)和类(class)的名字,默认值为CODE、DATA、BSS。

• Linker(连接器)