Turbo Assembler
汇 编 大 全

（上册）

用户手册

中国科学院希望高级电脑技术公司

一九九〇年九月

Turbo Assembler 汇编大全

用户手册

（上册）

汪亚文　译
文　韬　校

中国科学院希望高级电脑技术公司
一九九〇年九月

■ 北京市新闻出版局

准印证号：3205—90205

■ 订购单位：北京8721信箱资料部
■ 邮　　码：100080
■ 电　　话：2562329
■ 地　　址：海淀影剧院北侧
■ 乘　　车：320、302、332路车在海淀黄庄下车
■ 办公地点：希望公司大楼101房间

前言

Turbo Assembler 是 Borland 公司推出的可与 MicroSoft 公司的 MASM 相匹敌的汇编语言软件。从某种角度来看，Turbo Assembler 的总体性能要比 MASM 好。Turbo Assembler 的速度是 MASM 所远不能比拟的。Turbo Assembler 与 MASM 完全兼容，带有可选的 Ideal 模式。

Turbo Assembler 支持 Turbo Debugger 源级调试器，方便、高效的调试工具会帮助你减轻调试汇编原会有的负担。汇编的繁琐你将不再会感觉到，代之以汇编的高速、对计算机的完全控制。

Turbo 系列高级语言的使用者若想优化代码、提高速度、使用低级功能，请使用 Turbo Assembler，因为 Turbo Assembler 使 Turbo 系列高级语言与汇编的接口简化。

<<Turbo Assembler 大全>>分成三个部分：<<Turbo Assembler 用户手册>>，<<Turbo Assembler 程序设计技巧>>，<<Turbo Assembler 参考手册>>。

本书介绍高级的 Turbo Asembler 程序设计技巧。阐述 Turbo Assembler 与 Turbo C、Turbo Pascal、Turbo Prolog、Turbo Basic 的接口，80386 汇编程序设计，Ideal 模式的用法。

由于编译者水平有限，加上时间仓促，错误难免，欢迎指正。

文韬

1990.8.

内容简介

欢迎使用 Borland 公司开发的一次性扫描汇编器 Turbo Assembler。Turbo Assembler 具有超前引用特证，其汇编速度可达每分钟 48000 行(在 IBM PS/2 60 模式下)。它与 MASM 兼容，并带有可选的扩展语法的 Ideal(理想)模式。无论是初学者还是有经验的编程者都会欣赏这些特征及其提供的便于汇编语言程序设计的其它特征。以下仅列出其中的一些优点，在本书的后续章节中将有更详细的介绍:

* 全 80386 支撑环境
* 改进的语法类型检查
* 简化的段指令
* 改进的列表控制
* 局部标号
* 局部的栈符号及过程参数调用
* 结构和联合
* 嵌套指令
* 模拟 MASM 的 Quirks 模式
* Turbo Debugger 的全源调试输出
* 内部交叉引用工具(TCREF)
* 配置文件和命令文件

Turbo Assembler 是一个功能强大的命令行汇编器，它接受用户的源文件(.ASM)并输出目标模块(.OBJ)。用户可使用 Borland 公司开发的高速链接程序 TLINK.EXE 对目标模块进行链接并生成可执行文件(.EXE)。

Turbo Assembler 在 80x86 和 80x87 序列处理器环境下工作。(有关 80x86/80 x87 序列指令集的更详细的内容，请参看 Intel 公司出版的有关书籍。)

§0.1 硬件和软件需求

Turbo Assembler 在包含 XT、AT、PSA/以及所有兼容机在内的 IBM PC 序列机上运行。它要求 MS-DOS 2.0 或更高版本的操作系统，并至少配备 256K 内存。

Turbo Assembler 产生的是 8086、80186、80286 及 80386 指令代码。它也可产生 8087、80287、80387 数字协处理器的浮点数指令。

§0.2 有关本手册

Turbo Assembler 参考资料共有两本：<<Turbo Assembler 用户指南>>(本书)和<<Turbo Assmbler 参考指南>>。用户指南提供使用 Turbo Assembler 所需的基本指令及汇编语言程序设计的完备检查。参考指南介绍 Turbo Assembler 使用的算符、预定义符及伪指令。

下面详细介绍用户手册所包含的内容。

§0.2.1 用户手册

第一章: "初启" 介绍文件在盘上的分布及安装 Turbo Assembler 的操作流程。

第二章: "Turbo Assembler 程序设计" 引导用户用汇编语言进行程序设计, 其中包含有一些关于使用命令行开关的程序样例。该章也对常用计算机及特殊 8088 处理器进行了讨论。

第三章: "命令行参考" 详述所有的命令行选择项, 告知用户如何使用配置文件和命令文件。

第四章: "汇编程序的基本元素" 描述汇编语言的基本元素, 其中包含有关伪指令、指令、内存存取、段及其它固有信息。

第五章: "中级 Turbo Assembler 程序设计" 在第四章的基础上更进一步讨论 Turbo Assembler 中更高级的有关方面——关于伪指令、串指令等的更丰富的内容。本章也涉及到汇编程序设计过程中可能遇到的常见错误。

第六章: "Turbo Assembler 与 Turbo C 的接口" 描述如何联合使用汇编语言和一种高级语言, 即 Turbo C。本章将详细介绍如何把汇编语言模块链接到 Turbo C 中以及在 Turbo C 中如何调用 Turbo Assembler 函数。

第七章: "Turbo Assembler 与 Turbo Pascal 的接口" 讲述如何链接汇编代码和 Turbo Pascal 代码; 该章也举出了一些程序样例。

第八章: "Turbo Assembler 与 Turbo Basic 的接口" 介绍如何联合使用 Turbo Assembler 和 Turbo Basic, 其中有丰富的程序样例。

第九章: "Turbo Assembler 与 Turbo Prolog 的接口" 介绍如何联合使用 Turbo Assembler 和 Turbo Prolog, 其中包含有程序样例。

第十章: "Turbo Assmbler 高级程序设计" 详细介绍前几章中涉及到的内容, 如段前缀、宏、段伪指令等等。

第十一章: "80386 及其它处理器" 介绍 80386 程序设计。

第十二章: "Turbo Assembler Ideal 模式" 讲述有关 Ideal 模式的知识及使用 Ideal 模式的原因。

参考文献: 列出汇编程序设计时可参考的一些有用的书目。

§0.3 符号约定

谈及 IBM PCs 或其兼容机时, 指使用 8088、8086、80186、80286 和 80386 芯片(所有这类芯片常称作 80x86)的任何机型。涉及 PC-DOS、DOS 和 MS-DOS 时, 指 2.0 版本或更高版本的操作系统。

目录

第一章　初启

在引导用户提高汇编语言程序设计速度之前，用户务必完成以下事项。取出 Turbo Assembler 盘并为每张盘创建一个拷贝(用 DOS 命令)，以形成用户自己的“工作”拷贝。在此之后可保存好原程序盘。(交换用户损伤的软盘时有一定的费用开销，所以只使用原盘形成备份和工作拷贝。)

如果希望用 Turbo Assember 代替 MASM，那么可参看附录 B，以了解 Turbo Assember 与 MASM 的不同之处。

注意：使用 Turbo Assembler 之前务必阅读 README 文件，该文件囊括了有关程序及纠错和(或)手册附加内容的最新信息。

§1.1　软盘上的文件

* TASM.EXE: Turbo Assembler (Turbo 汇编器)
* TLINK.EXE: Turbo Linker (Turbo 链接器)
* MAKE.EXE: 命令行 MAKE 具
* TLIB.EXE: Turbo 程序库
* README.COM: 显示 README 文件的程序
* README: 有关软件和文书文件的详细信息
* TCREF.EXE: 源文件交叉引用工具
* OBJXREF.COM: 目标文件交叉引用工具
* GREF.COM: 过滤工具
* TOUCH.COM: 文件修改工具
* INSTALL.EXE: 安装程序
* MMACROS.ARC: MASM 模式宏的存档文件

§1.2 安装 Turbo Assembler

INSTALL 盘上有一个名为 INSTALL.EXE 的程序，它将有助于用户安装 Turbo Assembler。有两种可选择的安装方式：

1. 硬盘用户：　该可选项允许用户选择装载文件的子目录。

2. 软盘用户：　该可选项将使用 Turbo Assembler 时所需要的文件安装到双软盘驱动器系统中。在开始安装前务必准备四张已格式化的软盘。

要开始安装时，将当前驱动器改换到含有 INSTALL 程序的驱动器并打入 INSTALL。对屏幕底框中出现的每一提示，系统都给出相应的说明。例如，如果在 A 驱动器中安装 Turbo Assembler，则打入

```
INSTALL
```

在安装 Turbo Assembler 之前，务必阅 读 README 文件以得到与该方面有关的更进一步的信息。

注意： 如果希望在便携式电脑或使用 LCD 显示器的任何其它系统中运行 INSTALL 程序，那么在运行 INSTALL 之前应当将系统设置成黑白模式。可在 DOS 环境下用下列命令实现上述功能：

```
Mode  bw80
```

用户亦可使用/b 开关强制 INSTALL 在黑白模式下工作：

```
INSTALL /b
```

第二章　Turbo Assembler 程序设计

如果用户以前从未使用汇编语言设计程序，那么可从此处开始阅读。你可能听说过，汇编语言程序设计是一种只适于训练有素者或只适于天才的“未知项”。切勿相信! 汇编语言只是机器本身具有的一种高级语言。正如你所希望的一样，计算机语言是高度逻辑化的语言；也正如你可能希望的一样，汇编语言具有极其强大的功能——事实上，也只有汇编语言才能触及到 IBM PC 及其兼容机系列的核心处理器——Intel 80x86 系列的全部功能。

仅用汇编语言就可书写出完整的程序。用户也可将汇编语言嵌插在用 Turbo C、Turbo Pascal、Turbo prolog、Turbo Basic 及其它语言编写的程序中。无论在哪 种方式下，都可用汇编语言写出灵活快速的小型程序。汇编语言代码对计算机各方面操作的控制能力与其执行速度有同样的重要性，它可低级控制到系统时钟的最近一次变化。

本章介绍汇编语言并剖析汇编语言程序设计的独特优点。首先，用户将运行一些汇编语言程序，以便获得对该语言的感性认识，并逐渐习惯使用汇编语言进行编程。其次，本章将涉及常用的计算机及特殊的 8086 处理器的一些特点，以便用户了解 8086 汇编程序的特殊功能。本章也讨论了汇编语言程序设计中尤其与 IBM PC 有关的一些问题。

第四章“汇编程序的基本元素”是本章的后续内容，讲述了汇编语言程序的结构和程序的基本元素，并用完整的程序样例总结了这两章的内容。

第五章“Turbo Assembler 程序设计”继续介绍汇编语言程序设计的知识，第十章“Turbo Assembler 高级程序设计”进一步涉及到内存模式、宏及其它高级方面。

很显然，这几章的讲解并不能使用户成为汇编语言程设计专家；而只能引导用户熟悉汇编语言并开始编制自己的汇编语言程序。建议用户阅读有关汇编语言程序设计和 PC 机结构方面的专门书籍(请参看本书末尾的参考章节)。另外，IBM 公司的《DOS Technical Reference》、《BIOS Interface Technical》和《Personal Computer XT Technical Reference》三本手册是极有用的参考资料；这些手册讲述了汇编语言与 IBM 个人机的系统软件及硬件的接口。

进一步阅读之前，用户可能希望阅读第三章“命令行参考”，以熟悉命令行选择项。如果还没有安装 Turbo Assembler，则应该按照第一章“初启”的介绍安装之(使用 Turbo Assembler 盘的工作拷贝，或将文件从 Turbo Assembler 盘拷贝到硬盘上)。

最后一点：汇编语言是一种复杂的语言，要编写甚至是相对而言很简单的汇编语言程序也需要了解许多知识。有时在例子中使用了尚未介绍的语言特征，也是出于上述原因。在后续章节中将解释所有的语言特征。无论何时，只要用户对某一特征感到迷惑不解，就可查阅参考指南的第三章“指令”部分。

阅读了这些说明并准备好了第二卷第三章之后用户就可以创建自己的第一个汇编语言程序了。

§2.1　编写第一个 Turbo Assembler 用户程序

在程序设计领域里，第一个程序通常是一个显示信息的程序“Hello，World”，这是一个最好的起点。

```
.MODEL small
.STACK 100h
.DATA
HelloMessage DB 'Hello, World',13,10,'$'
.CODE
mov  as,@data                    ;set DS to point to the data segment
mov  dx,ax                       ;DOS print string function
mov  ah,9                        ;point to "Hello, world"
mov  dx.OFFSET HelloMessage      ;display "Hello, world"
int  21h                         ;DOS terminate program function
mov  ah,4ch                      ;terminate the program
int  21h          END
```

在输入 Hello.ASM 程序之后，将它存入磁盘。

熟悉 C 或 Pascal 语言的用户可能会想到，用以显示“Hello, World”的汇编语言程序看上去似乎稍长了一些。相对而言，汇编程序的确较长，因为每条汇编指令本身比 C 或 Pascal 指令完成的功能要少。另外，用户在组织这些汇编指令时有很大的灵活性，这意味着与 C 或 Pascal 不同，汇编语言可让用户对计算机编程，以让计算机做它力所能及的任何事情——要打入一些附加行时，这往往很有价值。

§2.1.1 汇编第一个用户程序

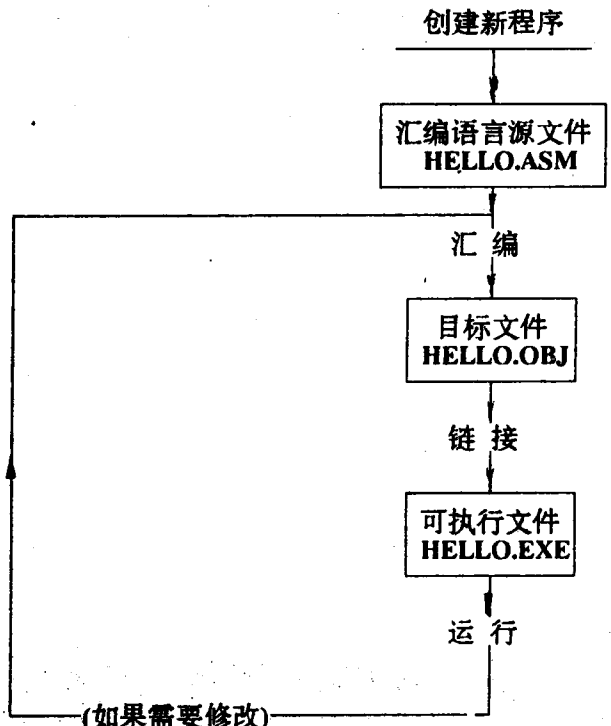

图 2.1 编辑、汇编、链接和运行循环

既然已保存了 HELLO.ASM 文件，用户可能希望运行该文件。但只有将它转换成可执行的形式(可以运行或执行)后，该程序才可运行。这就要求要有两个附加的步骤，即如图 2.1 所示的汇编和链接。图 2.1 描述了编辑、汇编、链接和运行程序的循环开发过程。

通过汇编这一步可将源代码转换成称之为目标模块的中间形式，通过链接可将一或多个目标模块组合成一个可执行的程序代码。可在命令行中完成汇编和链接工作。

要汇编 HELLO.ASM，可打入

```
TASM hello
```

如果不指明其它文件名，HELLO.ASM 将被汇编成 HELLO.OBJ。(注意，用户不必打入文件扩展名；Turbo Assembler 会将.ASM 作为扩展名。)用户可看到屏幕上出现下列内容:

```
Turbo Assembler Version 1. Copyright (C) 1988 by Borland International, Inc.
Assembling file: HELLO.ASM
Error message: None
Warning message:None
Remaining memory: 266K
```

如果完全按上述 HELLO.ASM 的内容准确地进行输入，那么屏幕上不会出现任何警告和错误。出现警告和错误时，警告和错误信息会显示在屏幕上，同时还会指出出错行的行号。如果出现错误，可检查程序代码并使之与上述内容完全一致，然后再次对其进行汇编。

§2.1.2 链接第一个用户程序

在成功地汇编了 HELLO.ASM 之后，与运行第一个汇编程序就只有一步之隔了。一旦链接了刚汇编过的目标模块并使之成为可执行的代码形式，用户就可以运行此程序。

要对程序进行链接，可使用 Turbo Assembler 自带的链接器 TLINK。在命令行中打入

```
TLINK hello
```

同样没有必要打入文件扩展名；TLINK 假定其扩展名为.OBJ。完成链接后(同样，至多需要几秒钟的时间)，链接器自动以目标文件的名字命名.EXE 文件，除非用户规定其它的名字。如果链接成功，屏幕上出现如下信息:

```
Turbo Linker Version 2.0 Copyright (c) 1987, 1988 by Borland
International, Inc.
```

链接过程中可能会出现错误，但本程序样例在链接时不会出错。如果用户看到错误信息(出现在屏幕上)，可修改源代码，使之与上述内容完全一致，然后重新汇编和链接。

§2.1.3 运行第一个用户程序

现在可以运行第一个用户程序了。在 DOS 提示符下打入 hello，这时信息

```
Hello, World
```

将被显示在屏幕上，并且这是程序所做的全部工作----你已经创建并运行了你的第一汇编语言程序!

§2.1.4 发生了什么?