

8086/8087/8088宏汇编
操作说明
用于8086开发系统

手册号121628-001

第二十一册

8086/8087/8088宏汇编
操作说明
用于8086开发系统

手册号121628-001

第二十一册

翻 译	顾维伦	(六二八所)
技术审校	钱承德	(计算机总局)

航空工业部第五七四厂

前 言

本手册的阅读对象是已熟悉8086/8087/8088宏汇编语言参考手册、已编好该语言的程序并已准备运行8086/8087/8088宏汇编程序的人员。本文内含有出错信息以及如何从造成这些错误的状态中恢复。下面所列的为与本文件有关并可能有用的参考资料：

Intellec系列Ⅲ微计算机开发系统程序员参考手册	121618
8086/8087/8088宏汇编语言参考手册(用于8086开发系统)	121627
iAPX86、88系列服务程序用户指南(用于8086开发系统)	121616
8086系列用户手册	9800722
8086系列用户手册数值计算补充	121586
MCS-86汇编语言转换程序使用说明书(用于ISIS-II用户)	9800642
Intellec系列Ⅲ微计算机开发系统控制台使用说明书	121609
PL/M-86用户指南(用于8086开发系统)	121636

本手册通过对汇编程序控制的使用，使用户学会使用8086/8087/8088宏汇编程序。正是根据这些控制或汇编程序的命令，汇编程序建立起一个目标文件和一个列表文件。

目 录

第一章 在使用 8086/8087/8088 宏汇编程序之前	(1)
第二章 调用 8086/8087/8088 宏汇编程序	(3)
第三章 确定汇编的条件	
控制说明	(5)
控制	(5)
缺省控制	(6)
显式控制	(7)
列表工作方式	(8)
宏调用和控制识别	(9)
汇编程序全部控制一览表	(10)
第四章 列表文件和出错打印文件	
列表文件	(13)
标题	(13)
程序体	(14)
单元 (LOC)	(15)
目标(OBJ)	(16)
相等(EQUATE)	(16)
包含(INCLUDE)嵌套指示符	(17)
行	(17)
宏嵌套指示符	(17)
源文本	(17)
符号表	(18)

名称.....	(19)
类型.....	(19)
值.....	(19)
属性.....	(20)
相互对照 (XREFS)	(20)
出错打印文件.....	(21)

附录A 出错信息和恢复

控制台出错信息.....	(23)
源文件出错信息.....	(24)
宏功能出错信息.....	(59)
控制出错信息.....	(63)

附录B 链接8086/8087/8088汇编语言和PL/M—86的程序

概述.....	(70)
适用于所有计算模型的条件和规定.....	(70)
堆栈用法规定.....	(70)
寄存器用法规定.....	(71)
PUBLIC和EXTERNAL的限制.....	(73)
各种计算模型特殊的条件和规定.....	(73)
过程和标号定义的规定.....	(73)
SMALL和COMPACT计算模型.....	(73)
MEDIUM和LARGE 计算模型	(74)
数据定义规定.....	(75)
局部定义数据(包括PUBLIC).....	(75)
SMALL、COMPACT和MEDIUM模型	(75)
LARGE模型	(76)
外部定义数据.....	(76)
SMALL、COMPACT和MEDIUM模型	(76)
LARGE模型	(77)
列表示例.....	(79)

附录C 程序实例: 向阴极射线管发送字符..... (101)

附录D 缩短控制的规则..... (102)

附录E 应用8087数字数据处理器和8087仿真程序..... (103)

词汇表..... (105)

第一章 在使用8086/8087/ 8088宏汇编程序之前

如果你是首次使用8086/8087/8088宏汇编程序，检查你用的Intellec系列Ⅲ 微计算机开发系统是否具有下列项目，因为这是汇编程序工作所必需的：

带192K RAM 内存的系列Ⅲ开发系统

一个控制台设备，如阴极射线管式(CRT)或电传打字机式(TTY)

(至少)一个磁盘驱动器和控制器

ISIS-II，4.1版或更新的版本

RUN，1.0版或更新的版本

你可能会希望增加一个行式打印机和/或更多的磁盘驱动器于这个配置中，但这是一个供8086/8087/8088宏汇编程序工作的最低限度配置。有关微计算机开发系统如何使用的详细说明，参见Intellec系列Ⅲ微计算机开发系统控制台使用说明书。

之后，检查所需的软件8086/8087/8088宏汇编程序是否已在软盘上。另一重要的需检查的软件是再定位和链接软件LINK86和LOC86。如果你还使用8087数字数据处理器或8087仿真程序，则应检查正确的库文件是否具备。这些文件的完整描述参见附录E。

INTELLEC SERIES III
MICROCOMPUTER
DEVELOPMENT
SYSTEM

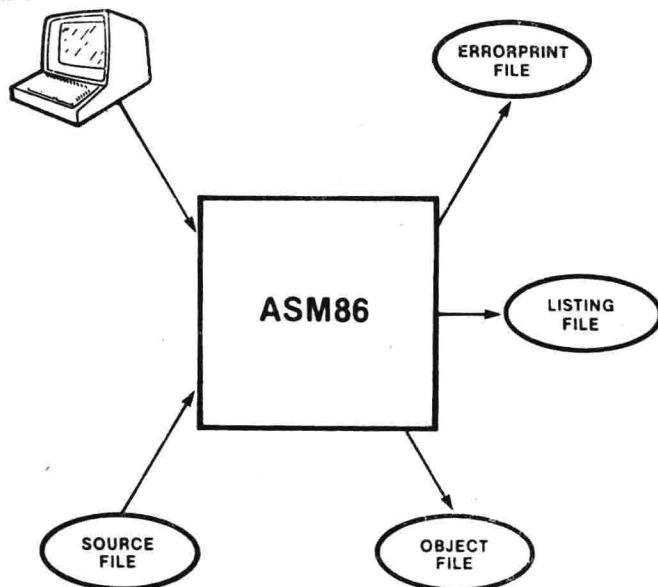


Figure 1-1. 8086/8087/8088 Macro Assembler Logical Files

121624-1

把8086/8087/8088汇编语言参考手册放在手边，因为该文件与本文件是互相有关的。这本手册假定读者已熟悉汇编语言，尤其是SEGMENT（段）和ASSUME（设定）伪指令。由于你至少会使用再定位和链接软件包的某些部份，有关该软件的手册也应放在手边，即iAPX86，88系列服务程序用户指南（供8086开发系统）。

本手册通过对其控制的使用教你如何使用8086/8087/8088宏汇编程序。正是按照这些控制，汇编程序建立起一个目标文件和一个列表文件。

这个列表文件包括你的源文件、扩展的宏源代码、汇编程序目标代码、汇编出错的一览表（如果存在）及在你的源程序中所定义的符号表。

8086/8087/8088宏汇编语言是一个多重复盖汇编程序。所有复盖都包含在一个文件内：ASM86.86。

汇编程序可驻留在任何磁盘上并从那里调用。

在汇编时，8086/8087/8088宏汇编程序建立6个临时文件供其使用，用户应了解这些文件。操作系统赋予这些文件以临时名称。这些文件在汇编结束时被注销。它们可以用WORKFILES（工作文件）控制放置于任何驱动器上。缺省条件则把这些文件放置在系统的工作文件驱动器上。

第二章 调用8086/8087/

8088宏汇编程序

调用8086/8087/8088宏汇编程序，应打入下列命令：

-[驱动器号:]RUN[驱动器号:]ASM86[驱动器号:]源程序

在上面格式中，连字符(-)为ISIS-Ⅱ操作系统的提示符，它表示操作系统已准备好接受命令（在这里是一个装入和执行汇编程序的命令）。短语[驱动器号:]在这个命令中出现三次，它表示当RUN程序、汇编程序和源文件的位置不是驱动器0时它们驻留的位置。方括号[]表示驱动器号在此命令中是一任选项。ASM86在命令行中必须存在，因为它是汇编程序本身的文件名。最后一项是待汇编程序的文件名。

上面的命令行调用汇编程序，此时汇编程序的缺省控制起作用。有关汇编程序控制及它们在汇编程序调用命令中的作用的详细介绍，参见第三章“确定汇编的条件”。第三章还将解释如何在另一行继续汇编程序调用命令。

有关RUN（运行）命令的使用的完整阐述，参见Intellec系列Ⅲ微计算机开发系统控制台工作说明书。

在你打入命令行之后，汇编程序立即发送一个开工信息至控制台：

SERIES-Ⅲ8086/8087/8088 MACRO ASSEMBLER, Vx.y

（系列Ⅲ 8086/8087/8088 宏汇编程序X.Y版）

其中X.Y是一个数1.0或更大。

当你的程序汇编完毕，结束信息和出错提要送至控制台，其形式如下：

ASSEMBLY COMPLETE, NO ERRORS FOUND

（汇编结束，没有发现出错）

如果汇编程序检测出错误，则出错提要（如下例所示）将出现：

ASSEMBLY COMPLETE, 1 ERROR FOUND

（汇编结束，发现1个出错）

如果汇编程序检测出严重出错，则源程序的处理提前终止，向控制台发送一个出错信息。附录A介绍各种严重出错和相应的控制台信息。

在初始使用阶段，你很可能会希望以上述方式使用汇编程序，因为这是一个缺省工作方式，即自动工作方式。随着你在使用汇编程序方面取得经验，第三章所阐述的汇编程序控制能使你在开发8086/8087/8088宏汇编语言程序方面把它用作一种更有效的工具。

你还应了解某些限制条件。由于大部分的限制条件是定量的，它们在本章中以表格

形式给出。

表2-1 8086/8087/8088宏汇编程序的参数（约定）

源文件：	
项目	数值
字符数/行	255个（包括 CR/LF ），如果打入更多字符，它们被处理但不示出，显示一个出错信息。
字符数/ID（标识符）	31个，如果打入大于31个，则大于部分被忽略。
符号数/模块	1200个（近似值），与名称的长度和符号的类型有关。
源行数/程序	汇编程序对此无限制。
连续行数/语句	汇编程序对此无限制。
字符数/字符串	255个（包括引号在内）
字符数/级别名	40个
重复嵌套	括号重数最多为8个。
PROC/SEG嵌套	16个（最多同时打开16个）。
项目数/PUBLIC、EXTRN、PURGE	汇编程序对此无限制。
项目数/GROUP（组）	36 SEG/GROUP
宏代码大小	汇编程序产生的代码（约）60个字节。
项目数/存储器初始化表格	项目数不超过16个，括号重数最多为8个。
SEGMENT大小	65535字节（64K-1）
记录极限	16个字段
记录大小	16位
列表文件：	
项目	数值
字符数/列表文件行	120个或PAGEWIDTH（页面宽度）所规定的最大值，如果更多的字符被输入，则产生循环复始*。
页面数/文件	汇编程序对此无限制
行数/文件	汇编程序对此无限制
内部：	
项目	数值
所需存储量	192K字节
中间文件大小	
I文件	1×源文件
S文件	30字节/符号
X文件	4字节/符号引用
M文件	随GEN的设置和宏功能的数目而变化，约2×源文件
L文件	约1×源文件
T文件	1.5×源文件

* 循环复始允许你输入比在PAGEWIDTH控制中规定的更多的字符或大于 120 个字符于一个列表文件行中；它把多余的字符放置在下一行中，最多为255个字符。

第三章 确定汇编的条件

控制说明

8086/8087/8088宏汇编程序共有12个主要控制，它们定义文件或标识设备，一般作为编程辅助手段。每一个控制是一个关键字，于调用8086/8087/8088宏汇编程序的同一行或在源文件的命令行内打入。在任何控制之后打入一个“&”可继续调用行。每一个控制可接通或断开，根据你所需的何种（任选的）输出和在那里你需要它而定。这些控制可根据你的需要输入任意次。汇编程序忽略每个控制除最后（最外层的）的定义以外的所有的定义（源文件的控制首先被扫描）。如果出现一个不是本章所提到的控制，则汇编程序中止工作，必须重新打入命令。ISIS—Ⅱ关于调用一个程序的行数限制(120)在这里也适用。参见Intellec系列Ⅲ微计算机开发系统控制台使用说明书（如果你还不熟悉的话）。

控制

控制有二类，即一般控制和主要控制。主要控制可用在请求调用行中或在源文件中任何非控制行之前；主要控制先被扫描直至达到一非控制行，即在其第一列没有“\$”的行，甚至空白行或注解行也将结束对主要控制的扫描。在含有主要控制的行内宏调用将不执行，实际上这将造成控制出错。

如果在调用行内发现出错，汇编即中止，ASM86报告一个控制出错（见第四章）。如果在源文件控制行中存在一个出错，处理仍将继续，一个出错信息放入列表文件和出错打印文件（如果有的话）中，而此后的所有控制行都将被忽略。

如果在源文件的前面几行说明了一个主要控制，在请求调用行也作了说明，则在请求调用行中说明的控制条件起作用。这是因为在源文件的主要控制比在请求调用行中的先被处理，而总是最后的设定被采用。例如，在下面的例子中，PAGING（分页）控制将起作用，尽管在源文件的第一行中说明了NOPAGING（不分页）。注意，这里利用“\$”继续请求调用行。

```
SERIES-Ⅲ 8086/8087/8088 MACRO ASSEMBLER VX.Y ASSEMBLY OF  
MODULE SAMPLE
```

```
OBJECT MODULE PLACED IN : F1 : SAMPLE. OBJ
```

```
INVOCATION LINE CONTROLS : PAGEWIDTH (72) SYMBOLS
```

```
PAGING
```

```
LOC    OBJ    LINE    SOURCE
```

```
1      $NOPAGING
```

```
2      NAME SAMPLE
```

一般控制是指这样的控制，它可以在源文件的任何行中说明，或在任何的INCLUDE（包含）文件、用户定义的宏功能或请求调用行中说明。一般控制或立即引起一个动作发生，如EJECT（换页），或起作用直到被另一个一般控制所取代（OVERRIDE），如LIST（列表）。在请求调用行中的一般控制比在源文件中的先处理，在源文件起始部分，一般控制说明的语法以及主要控制说明的语法如下：

\$CONTROL[（PARAMETER）][CONTROL[（PARAMETER）]]……

主要控制有：

OBJECT 或 NOOBJECT	（建立或不建立目标文件）
PRINT 或 NOPRINT	（建立或不建立列表文件，常称为表格文件或打印文件）
PAGELLENGTH, PAGEWIDTH	（设定垂直和水平页面范围）
PAGING 或 NOPAGING	（在上述的列表文件中产生或不产生页面编号）
NOSYMBOLS或SYMBOLS	（在列表文件的末端不附加或附加符号表）
NOERRORPRINT或ERRORPRINT	（向指定的文件不发送或发送出错信息表）
NODEBUG 或 DEBUG	（在目标文件中不放入或放入供再定位和链接软件用的局部符号信息；对于ICE操作，DEBUG是必需的）
NOXREF 或 XREF	（在源文件中，不给出每个符号位置相互对照或给出相互对照）
WORKFILES	（确定把临时的汇编程序工作中的文件置于规定的驱动器上）
MACRO 或 NOMACRO	（在源文件汇编过程中执行或不执行宏功能处理）
TITLE	（在列表标题处加一题目名称）

缺省控制

在上面的表中，左侧一系列的控制是缺省或自动工作方式（除PAGELLENGTH、PAGEWIDTH和WORKFILES外）。一旦汇编程序被调用，这些控制即起作用。如果你要求规定与这些不同的条件，你必须在命令中说明它们。

调用带控制的8086/8087/8088宏汇编程序的形式语法为：

—[：device：]RUN[：device：]ASM86[：device：]yourProgramname[CONTROL-LIST]

（—[：设备：]RUN[：设备：]ASM86[：设备：]你的程序名[控制表]）

记住，如果你选择不打入上面控制表中的任何控制，就意味着选用缺省控制，亦即它们是自动起作用的。也就是说表中左侧的控制起作用。但是，如果你选用上面的任何控

制，它必须按所示规定打入。表3-1阐述如何缩短这些控制的方法。控制间必须至少间隔一个空白字符。此外，如果控制有一个参数，则此参数必须置于括号之内。系列Ⅲ有关文件名称和设备名称的规定在这里也适用。如果你还不熟悉Intellec系列Ⅲ微计算机开发系统控制台使用说明书的话，请参阅该书。

下面是调用8086/8087/8088宏汇编程序的一个例子：

```
—RUN ASM86 LOOT.SRC OBJECT PRINT PAGING SYMBOLS
```

这个命令在驱动器O上产生一个目标文件LOOT.OBJ，建立一个带页码的列表文件LOOT.LIST，列表文件内含有符号表和出错信息。它等同于下面这个命令：

```
—RUN ASM86 LOOT.SRC SYMBOLS
```

这是由于缺省控制的自动或隐式调用的结果。注意，在缺省情况下，目标文件和列表文件都建立在与源文件相同的驱动器上。

显式控制

如果你希望定义与缺省不同的文件名称或设备，你必须在命令中说明。作为一个规定文件名称的例子，下面这个命令规定输出文件具有与输入文件不同的名称：

```
—RUN ASM86 LOOT.SRC OBJECT (:F1:COOT.OBJ)PRINT(COOT.LST)
PAGING
```

这样，你得到的目标文件和列表文件分别称为：F1:COOT.OBJ和COOT.LST。文件名称置于括号内。如果你包含一个文件名称而没有用括号，则命令就不能正确地读入。

小结：缺省操作就是调用汇编程序，产生一个目标文件并建立一个带页码的汇编列表文件。最紧凑的调用命令是：

```
—RUN ASM86 YOURPROGRAM(你的程序)
```

下面是这个命令最大可能的显式：

```
—[:device:]RUN[:device:]ASM86[:device:]yourprogram controls
```

在汇编你的程序的过程中，你常常还必需考虑时间和存储或内存空间，你可能会要求有缺省条件以外的灵活性。考虑到这个要求，你可以把在汇编过程中所建立的文件，在系统所允许的范围内，使之位于不同的设备或驱动器上。如果存储空间是一个主要问题，并且这是你的第一遍汇编，它可能会形成一个大的程序，这时你可能并不想在你的第一遍汇编中产生目标代码，这可由下面的命令来完成：

```
—RUN ASM86 LOOT.SRC NOOBJECT
```

采用这个命令，你可以检验你的程序而不浪费存储空间。下面是使缺省控制无效的另一例子：

```
—RUN ASM86 LOOT.SRC NOPRINT
```

这个命令抑制列表文件的建立。作为进一步的结果，它还使PAGING、SYMBOLS和XREF无效，

如果你希望目标代码驻留在与包含源程序驱动器不同的驱动器上，可用下面的命令：

```
—RUN ASM86 LOOT.SRC OBJECT (:F1:LOOT)
```

上面这个例子中的目标文件在驱动器1中，而源文件在驱动器0中。同样，你如果要求目的代码具有与源程序完全不同的文件名称，则必须予以规定：

—RUN ASM86 LOOT.SRC OBJECT (COOT)

如果你希望所产生的列表文件输出到与缺省（即与源文件同一设备）不同的设备，或如本例，至行式打印机：

—RUN ASM86 LOOT.SRC PRINT (:LP:)

这使得列表文件在行式打印机上打印输出。

有时候使列表文件具有与源文件不同的名称将会带来很大的方便，下面的命令起着这种作用：

—RUN ASM86 LOOT.SRC PRINT (:F1:SOOT.LST)

如果你想把汇编程序产生的临时文件放置在与包括源文件的驱动器不同的磁盘驱动器上，则必需用控制WORKFILES。此控制取二个驱动器名称作为其参数，前三个临时文件，即S、X和M文件放置在以第一个参数命名的驱动器上，而其余三个文件，即I、T和L文件则放置在以第二个参数命名的驱动器上。因此，要使工作文件产生在驱动器4和5上，应调用汇编程序如下：

—RUN ASM86 LOOT.SRC WORKFILES (:F4:, :F5:)

如果在源文件中规定了多于2个的WORKFILES，则汇编程序给出一个出错信息，把工作文件放置在所用的前二个驱动器上，其它驱动器则被忽略。如果在调用行内有二个以上的驱动器，即指示出错，处理行止。如果只规定一个驱动器，汇编程序把二组工作文件都放在指定的驱动器上。

如果你想要一张出错表而不要列表文件，可用如下命令形式：

—RUN ASM86 LOOT.SRC NOPRINT ERRORPRINT

小结：为“行止”或取代一个缺省控制，必须显式地包括进明显不同的控制、设备或文件名称。

列表工作方式

8086/8087/8088宏汇编程序有三个供用户使用的列表工作方式，每个都由一个一般控制所规定。这三个控制GEN、NOGEN和GENONLY，以各种不同的组合形式控制着源文件、宏调用和宏功能文本的列表。因为宏功能扫描程序是面向字符，而不是面向行的，宏功能调用可占据一行的一部分、整行或几行。宏功能的展开也可以占据几行、整行或一行的部分。这就是在GEN列表工作方式中，在已汇编的文件中有时会出现行的部分的原因。

规定GEN控制会产生这样的列表，它包括所有的源文本、所有宏调用和每个宏功能的展开（即宏功能文本）。展开是缩进若干字符以与宏调用上下对齐。每个宏调用的展开在调用行之下起开始打印。因此，GEN控制对宏调用和展开过程提供完整的示踪。所以，它适用于调试宏功能和得到最完整而连续的源文件的列表。但是，在包含许多宏调用的程序内，GEN控制会产生一个过分庞大的输出。

规定NOGEN控制产生一个只包含源文件文本的列表，例外的情况是，宏功能展开

行出错时将列表输出, 同时伴随相应的出错信息。与含有宏调用的行一起, 第一展开行 (如果有) 的目标代码被列表。NOGEN控制产生一个仅包含向宏处理程序输入 的列表。因此, NOGEN使源文件的编辑容易, 但宏功能被忽略。

规定GENONLY作为列表的控制产生这样的列表, 它仅包含源文件非宏文本 和 所调用的所有宏功能的文本, 这个控制忽略所有宏调用的列表。GENONLY产生的 列表只包括宏处理程序的输出。GENONLY对调试整个程序是有用的, 因为所有的目标 代码 (仅是目标代码) 皆示出。但复杂的宏功能可能会引起问题。

行的编号在GENONLY和NOGEN中是相同的。在 NOGEN 工作方式中, 行的 编号将会跳过在源文件文本中出现宏功能的区域。所跳过的行数准确地相等于 如果 使用 GENONLY所产生的行数。因此, 在NOGEN产生的列表文件中行的编号将与由GEN- ONLY所产生的列表文件中的同一行相应。

宏调用和控制识别

本节讨论控制和宏处理的相互作用。它只适用于一般控制行, 因为在源文件开始部 份主要控制行中的宏功能不能被识别。

通常, 在源文件中出现的控制行被识别并立即被处理。但是, 在宏处理中出现几种 情况, 它需要扩展控制处理的这一基本概念。宏调用应能产生整个控制行, 或控制行 的一部分。对提供条件产生控制行, 尤其是INCLUDE, 特别重要。为此, 应能在宏功能 定义中 (或在IF、WHILE或REPEAT的体内) 放入控制行, 并延迟控制行至 宏 功能 被调用 (或IF、WHILE或REPEAT被展开) 时才被识别和执行。

这一机理通过链接控制行的扫描和宏处理程序的二种扫描方式 (“正常” 和 “文 字” 扫描) 而达到。请回想一下, 在 “正常” 扫描方式, 所有宏调用被识别和展开; 在 “文字” 扫描方式, 所有宏调用不被识别, 而是作为普通的文本字符串 而通过。 “文 字” 扫描的几个例子为: % * DEFINE功能的体、用 “% * ” 调用的用户宏功能 的展 开和% (…) 功能的内部等。下文还将给出更多的例子。

在 “正常” 方式下被扫描的控制行被识别和处理。在 “文字” 方式中被扫描的控制 行不被识别。因为一行的各个部份可以不同方式被扫描, 实际的控制识别方式系当控制 指示符 “\$” 被 扫描 到的扫描方式。如果 “\$” 是在正常方式下被扫描的, 则该行的 其余部份作控制行处理。如果 ‘\$’ 是在文字方式下被扫描的, 则 ‘\$’ 以及 该行的其 余部份作一般的文本处理。下面给出在各个扫描方式下遇到的控制示例。

应该指出几个与控制有关的注意事项, 它们由控制行被扫描的方式所引起的。首先 是在控制行末端的换行 (LF) 必须处于与窗口 ‘\$’ 同一嵌套层上 (亦即 “向上” 调 用是不允许的)。其次, 在宏功能中的控制行使宏功能嵌套层次增加1。最后, 如果在 控制行中出现宏功能出错, 宏功能嵌套信息的查找包含一个控制项, 如同对 “\$” 的 “调用”。

1. 定义一个宏功能, 它的定义由另外一个文件包含 (INCLUDE) 入内:

```
%DEFINE (FOO) (  
$INCLUDE (:F1:DEFN.INC)
```

)

因为DEFINE被正常地调用(即用“%”而不是“%·”),定义体以“正常方式”被扫描。因而\$ INCLUDE控制行被立即识别,FOO被定义为INCLUDE文件的内容(DEFN.INC的内容被读入并以FOO的值的形式储存)。

如果FOO是用%·DEFINE以文字方式被定义,则INCLUDE可推迟到宏指令被调用:

2. 定义一个宏功能,在被调用时它包含一个文件:

```
%·DEFINE (FOO) (
$ INCLUDE ( : F1 : DFN.INC )
)
```

对文件的条件INCLUDE表明处理方式的重要性。

3. 对二个文件中的一个作条件INCLUDE:

```
IF ( Condition ) THEN (
$ INCLUDE ( ONE )
) ELSE
$ INCLUDE ( TWO )
) FI
```

THEN和ELSE二个子句都以文字方式被扫描,而只有一个被扩展(在这种情况下被选定的子句是正常地扩展,因为用了%IF而不是%·IF)。结果,二个文件中只有一个被包含入。在这一情况下,不能用%·IF。

4. 定义一个产生控制的宏功能:

```
%·DEFINE ( PRINT ( X ) (
$ %X% ( ) LIST
)
```

因为用了“%·DEFINE”,控制行在定义点上没有被处理,而是推迟到宏功能实际调用时才被处理:

宏调用

```
%PRINT ( )
```

产生出控制行

```
$ LIST
```

而

```
%PRINT (NO)
```

则产生如下控制行

```
$ NOLIST
```

汇编程序全部控制一览表

下表为8086/8087/8088宏汇编程序的全部控制及其缩写形式。

表3—1 8086/8087/8088宏汇编程序控制一览表

DEBUG DB	局部符号信息放置在目标文件中供符号调试用	主要
• NODEBUG NODB	局部符号信息不放置在目标文件中	
EJECT EJ	指示一个页面分开位置。在EJECT控制后的第一行开始新的页面。缺省是无页面断开。NOPAGING则取代EJECT控制。	一般
ERRORPRINT [(FILE)] EP	给出在汇编中所发生的出错的一览表	主要
• NOERRORPRINT NOEP	不给出所发生的出错的一览表	
GEN GE	在列表文件中提供在各层次的宏调用、宏扩展的完整列表。宏调用后的扩展出现在一个新行，向里凹与调用的%对齐。	一般
NOGEN NOGE	仅提供源文件各行的列表，不打印宏功能体，但有错误的宏功能扩展行例外。	
• GENONLY GO	提供完全扩展的源文件的列表。这是缺省控制。	
INCLUDE(;Fn: FILENAME) IC	INCLUDE控制其后的源文件行从指定的文件输入。输入一直持续到检测到文件结束代码为止。这时，输入又重新恢复到INCLUDE控制出现时正在进行处理的那个文件。如果INCLUDE出现在请求调用行中，则所包含进的文件将插入在主源文件之前。如果INCLUDE出现在主要控制行中，则所包含进的文件将插入在该主控制行之后。在同一时间，起作用的宏调用和INCLUDE控制的组合不应超过54个。	一般
• LIST LI	在LIST控制后，使汇编程序在源文件列表中包含列表行	一般
NOLIST NOLI	使汇编程序抑制列表行的打印直到再次读入LIST控制为止	一般
• MACRO MR	通知汇编程序应处理宏功能。	主要
NOMACRO NOMR	允许用户对已包含的宏功能在源文本中不处理以节省汇编时间。	
• OBJECT((FILE,EXT)) OJ	产生目标代码文件并输出至指定的磁盘文件上。缺省是使目标文件具有与源文件相同的文件名称而带一个扩展项OBJ，并驻留在与源文件同一磁盘上。	主要
NOBJECT NOOJ	不产生目标代码。	
• PAGELENGTH (n) pL (n)	表示列表文件每页的行数。缺省是60行。	主要

• PAGEWIDTH (n) • PW (n)	设定列表文件每行所允许的最大字符数。超出的字数则“循环复始”至下一行。这一控制使用户可设定循环复始的极限。缺省是120个字符/行，最小 60，最大 255。	主要
PAGING PI	列表文件以带页码的页面构成，每页的顶部有标题。如果在命令行中有 NOPRINT ，则 PAGING 被抑制。	主要
NOPAGING NOPI	列表文件没有格式页面。符号表与源文件相隔4行。	
• PRINT(FILE,EXT) PR	产生列表文件，输出至指定的文件或设备。如果在命令行中既没有 PRINT ，也没有 NOPRINT ，则列表文件具有与源文件相同的名称(带扩展项 LST)，驻留在与源文件同一个磁盘上	主要
NOPRINT NOPR	列表文件被抑制。其结果隐含着 NOPAGING 、 NOSYMBOLS 和 NOXREF 。	
SAVE.....RESTORE	这二个控制允许使某些一般控制的设定在 INCLUDE 控制把输入源转接至另一文件之前被保存在堆栈上，在被包含的文件结束时再恢复。但 SAVE 和 RESTORE 也可用于其它目的。可保存和恢复设定的控制有： LIST/NOLIST 和 GEN/NOGEN/GENONLY 。 SAVE 的最大嵌套层次为8。	一般
SYMBOLS SB	列表文件的末尾带有符号表。如果规定 SYMBOLS ，则在列表文件的末尾出现一张以字母顺序排列的符号表。如果在命令行中有 NOPRINT ，则 SYMBOLS 被抑制。	主要
• NOSYMBOLS NOSB	在列表文件的末尾不带有符号表。	
• TITLE() TT	允许用户把标题赋与一页或若干页。括号必须成对，引号与任何其它字符同样处理。缺省是程序块的名称。如果 TITLE 控制出现在源文件的开头或请求调用行内，则标题打印在第一页的顶部。如果标题含有在系列Ⅲ的控制中有效的任何分隔符，则它们必须用单引号括起来，否则 ASM86 会报告一个出错。关于有效的分隔符列表，参见 Intellec 系列Ⅲ微计算机开发系统控制台使用说明书，缺省标题是由 NAME 规定的程序块名称。	一般
• WORKFILES(:FX: , :FY:) WF	汇编程序产生的临时文件放置在规定的磁盘或设备上。缺省则把临时文件放置在系统工作设备上。	主要
XREF XR	提供一张符号表，它带有用户定义的符号被定义、引用和删除的对应行的相互对照。缺省是 NOXREF ， XREF 也取代 NOSYMBOLS 。	主要
• NOXREF NOXR	在列表文件末尾处的符号表内不放置相互对照信息。	

• 为缺省控制