

IBM 多功能微机

5550

MACRO

宏汇编用户指南

刘 滨 译

祝 军 校

201378

中国科学院沈阳分院
科技咨询开发服务部

MACRO

宏汇编用户指南

刘 滨 译
祝 军 校

中国科学院沈阳分院
科技咨询开发服务部

前 言

美国IBM公司推出的IBM5550微机已被我国许多部门所应用。由于随机所带的外文资料有限，中文资料不足，为满足广大用户的迫切需求，推动IBM5550的开发与应用，我们组织了该机外文资料的翻译。

此套中文资料是从IBM公司最新原文版本直接翻译而成，保持原文基本精神和全部内容。译前曾征求了一些用户意见，现译出二十册，供各方面用户选择使用。为适合我国实际情况，对原文不易理解之处做了必要的注释和补充，有的地方进行了必要的整理，以尽量适用于用户的要求。

本资料是由中国科学院沈阳分院王生蒂同志主持，中国科学院沈阳计算技术研究所王凤文同志编辑。参加翻译工作的有中国科学院沈阳计算技术研究所、辽宁省计委计算中心等单位。译者都是从事计算机应用开发和研究并有较好的外语水平和翻译实践经验的科技人员。他们为翻译出版此部资料做了大量的有益工作，在此一并致以谢意。

由于时间仓促，水平有限，错误和不妥之处在所难免，敬请用户批评指正。

**中国科学院沈阳分院
科技咨询开发服务部**

一九八五年六月

前 言

本书是一本面向使用 IBM 多功能 5550 MACRO 汇编程序的有经验的汇编语言程序员的参考手册。

手册中就如何使用 MACRO 汇编程序、交叉引用程序，伪操作及机器指令给出了详细说明。

手 册 中 的 内 容 编 排

- 第一章** “绪论”。对 MACRO 汇编程序及特性作了简要地叙述。
- 第二章** “MACRO 汇编”。详细地叙述了汇编、连接和运行程序时所需要的各种命令。
- 第三章** “交叉引用程序”。对用于源程序的调试和列表的功能作了描述。
- 第四章** “汇编语言的格式”。解释汇编语言的语句、数据的表示、寄存器的用途以及有效地使用 MACRO 汇编程序所应了解的一些其他内容。
- 第五章** “伪操作”。介绍怎样简化编码、减少生产错误的原因及保证对指令的标准顺序进行编码。
- 第六章** “指令记忆码”。描述 MACRO 汇编程序处理的各种指令，解释每条指令的功能和怎样对这些指令进行编码。本章中还给出了指令是如何工作的例子。
- 附录 A** “错误信息”。列出 MACRO 汇编程序产生的各种错误信息。
- 附录 B** “指令表”。给出 MACRO 汇编程序指令的简要说明。
- 附录 C** “伪操作表”。提供数据、条件、宏和列表伪操作的一览表。
- 附录 D** “会说实例”。介绍有关汇编、连接和运行汇编程序软盘上实例程序的过程。
- 词 汇** “词汇”。定义在本手册中可能碰到的各种术语。

译 丛 目 录

1. 系统概况
2. 系统安装
3. 系统操作指南
4. 系统维修手册
5. MACRO宏汇编用户指南
6. BASIC用户指南
7. BASIC编译程序
8. FORTRAN编译程序
9. COBOL编译程序
10. PASCAL编译程序用户指南
11. DOS用户指南
12. Wordstar字处理程序用户指南
13. Wordstar教学手册
14. Spellstar用户指南
15. Multiplan
16. Multitool Chart用户指南
17. Multitool File用户指南
18. Mailmerge用户指南
19. 汉字输入方法
20. 汉字编码手册

目 录

第一章 绪论	(1)
机器指令.....	(1)
汇编程序指令.....	(1)
宏指令.....	(2)
伪操作.....	(2)
EDLIN 程序.....	(2)
处理器的结构.....	(2)
编址.....	(2)
寄存器.....	(3)
栈.....	(3)
第二章 MACRO 汇编程序	(4)
命令格式.....	(4)
隐含的文件名扩展.....	(4)
怎样启动汇编程序.....	(5)
如何用选择 1 启动汇编程序.....	(5)
如何用选择 2 启动汇编程序.....	(6)
如何用选择 3 启动汇编程序.....	(7)
MACRO 汇编程序的结束.....	(8)
参数 (/Parms)	(8)
设备标识符.....	(9)
第三章 交叉引用程序 (CREF)	(10)
建立交叉引用文件.....	(10)
怎样启动CREF.....	(10)
如何用选择 1 启动CREF.....	(10)
如何用选择 2 启动CREF.....	(11)
如何用选择 3 启动CREF.....	(12)
CREF的结束.....	(12)
CRF 的格式.....	(12)
带有开始记录类型的记录.....	(12)
带有结束记录类型的记录.....	(12)
第四章 汇编语言的格式	(13)
符号/数据表示法	(13)
数据项.....	(13)

常数 (立即值)	(14)
第一次扫描与第二次扫描	(14)
标号	(16)
变量	(16)
寄存器代码的约定	(17)
标志寄存器	(17)
操作数	(18)
属性操作符	(19)
带有返面值的操作符	(20)
记录专用操作符	(23)
表达式	(23)
第五章 伪操作	(25)
数据伪操作	(25)
ASSUME	(25)
COMMENT	(26)
DB 定义字节	(26)
DD 定义双字	(27)
DQ 定义四倍长字	(28)
DT 定义十个字节	(29)
DW 定义字	(30)
END	(31)
EQU	(31)
= (等号)	(32)
FVEN	(32)
EXTRN	(32)
GROUP	(33)
INCLUDE	(35)
LABEL	(35)
NAME	(36)
ORG	(37)
PROC	(37)
PUBLIC	(38)
·RADIX	(39)
RECORD	(39)
SEGMENT	(41)
STRUC	(43)
条件伪操作	(44)
IF expression	(45)

IFE expression	(45)
IF 1	(45)
IF 2	(45)
IFDEF symbol	(45)
IFNDEF symbol	(45)
IFB <argument>	(45)
IFNB <argument>	(45)
IFIDN <argument-1>, <argument-2>	(45)
IFDIF <argument-1>, <argument-2>	(45)
ENDIF	(45)
ELSE	(45)
宏伪操作	(46)
宏和重复块伪操作	(46)
MACRO	(50)
PURGE	(51)
LOCAL	(51)
REPT	(52)
IRP	(52)
IRPC	(52)
ENDM	(53)
EXITM	(53)
& (连接号, 特殊的MACRO操作符)	(53)
; ; (两个分号, 特殊的MACRO操作符)	(54)
! (惊叹号, 特殊的MACRO操作符)	(54)
% (百分号, 特殊的MACRO操作符)	(54)
列表伪操作	(55)
.CREF/.XCREF	(55)
.LALL/.SALL/.XALL	(56)
.LIST/.XLIST	(56)
%OUT	(56)
PAGE	(57)
SUBTIL	(57)
TITLE	(58)
假条件块	(58)
.LFCOND	(58)
.SFCOND	(58)
.TFCOND	(58)
编程控制	(59)

第六章 指令记忆码	(61)
符号与表示法.....	(61)
指令字段.....	(63)
操作代码字节.....	(63)
寻址方式字节.....	(63)
AAA 加法ASCII 调整.....	(64)
AAD 除法ASCII 调整.....	(65)
AAM 乘法ASCII 调整.....	(65)
AAS 减法ASCII 调整.....	(66)
ADC 进位加.....	(66)
存贮器或寄存器操作数与寄存器操作数相加.....	(67)
立即操作数与累力器相加.....	(67)
立即操作数与存贮器或寄存器操作相加.....	(68)
ADD 加法.....	(68)
存贮器或寄存器操作数与寄存器操作数相加.....	(68)
立即操作数加到累加器.....	(69)
立即操作数加到存贮器或寄存器操作数.....	(69)
AND 逻辑与.....	(69)
CALL 调用过程.....	(71)
CBW 字节转为字.....	(73)
CLC 清进位标志.....	(73)
CLD 清方向标志.....	(74)
CLI 清中断标志.....	(74)
CMC 补码进位标志.....	(74)
CMP 比较两个操作数.....	(75)
CMPS/CMPSB/CMPSW 比较字节串或字串.....	(76)
CWD 字转为双字.....	(78)
DAA 十进制加法调整.....	(78)
DAS 十进制减法调整.....	(79)
DEC 减量为1的目的.....	(79)
寄存器操作数(字).....	(79)
存贮器或寄存器操作数.....	(80)
DIV 无符号除法.....	(80)
ESC 换码.....	(82)
HLT 暂停.....	(82)
IDIV 带符号的整数除法.....	(83)
IMUL 整数乘法.....	(84)
IN 输入字节或字.....	(85)

INC	增量为 1 的目的	(86)
INT	中断	(87)
INTO	溢出中断	(88)
IRET	中断返回	(89)
JA/JNBE	高于/不低于或不等于时转	(89)
JAE/JNB	高于或等于/不低于时转	(90)
JB/JNAE/JC	低于/不高于也不等于/有进位时转	(91)
JBE/JNA	低于或等于/不高于时转	(91)
JCXZ	CX 为零时转	(92)
JE/JZ	等于/零标志位为 1 时转	(92)
JG/JNLE	大于/不小于也不等于时转	(93)
JGE/JNL	大于或等于/不小于时转	(94)
JL/JNGE	小于/不大于也不等于时转	(95)
JLE/JNG	小于或等于/不大于时转	(95)
JMP	无条件转	(96)
	段内或组内的直接无条件转移	(96)
	段内直接短标号的无条件转移	(97)
	段间的直接无条件转移	(97)
	段间的间接无条件转移	(97)
	段内或组内的间接无条件转移	(97)
JNC	无进位时转	(98)
JNE/JNZ	不等于/零标志为 0 时转	(98)
JNO	无溢出时转	(99)
JNP/JPO	无奇偶校验/奇偶校验为奇时转	(99)
JNS	无符号/为正时转	(100)
JO	溢出时转	(100)
JP/JPE	有奇偶校验/奇偶校验为偶时转	(100)
JS	符号标志位为 1 时转	(101)
LAHF	把标志位装入 AH	(101)
LDS	装入数据段寄存器	(102)
LEA	装入有效地址	(102)
LES	装入附加段寄存器	(103)
LOCK	封锁总线	(104)
LODS/LODSB/LODSW	装入字节串或字串	(104)
LOOP	计数寄存器不为 0 时循环	(106)
LOOPE/LOOPZE	等于/零标志位为 1 时循环	(107)
LOOPNE/LOOPNZ	不等于/零标志位为 0 时循环	(108)
MOV	传送	(109)

累加器的内容传送到寄存器·····	(109)
存贮器的内容传送给累加器·····	(109)
存贮器或寄存器操作数的内容传送给段寄存器·····	(110)
段寄存器的内容传送给存贮器或寄存器·····	(110)
MOVS/MOVS _B /MOVSW 传送字节串或字串·····	(112)
MUL 无符号乘法·····	(113)
NEG 求反/形成二进制补码·····	(114)
NOP 空操作·····	(115)
NOT 逻辑“非”·····	(115)
OR 逻辑“或”·····	(116)
存贮器或寄存器操作数与寄存器操作数的逻辑“或”·····	(116)
立即操作数与累加器操作数的逻辑“或”·····	(117)
立即操作数与存贮器或寄存器的逻辑“或”·····	(117)
OUT 输出字节或字·····	(117)
固定端口·····	(118)
可变端口·····	(118)
POP 从堆栈弹出字并送给目的寄存器操作数·····	(118)
段寄存器·····	(119)
存贮器或寄存器操作数·····	(119)
POPF 标志位弹出堆栈·····	(119)
PUSH 把字压入堆栈·····	(120)
寄存器操作数(字)·····	(120)
段寄存器·····	(120)
存贮器或寄存器操作数·····	(121)
PUSHF 把标志寄存器压入堆栈·····	(121)
RCL 通过进位标志位循环左移·····	(121)
RCR 通过进位标志位循环右移·····	(123)
REP/REPZ/REPE/REPNE/REP _{NZ} 重复串操作·····	(124)
RET 过程返回·····	(125)
段内返回·····	(126)
段内返回并把立即值加到栈指针·····	(126)
段间返回并把立即值加到栈指针·····	(126)
ROL 循环左移·····	(127)
ROR 循环右移·····	(128)
SAHF AH装入标志寄存器·····	(129)
SAL/SHL 算术左移/逻辑左移·····	(130)
SAR 算术右移·····	(131)
SBB 带借位减法·····	(132)

SCAS/SCASB/SCASW	扫描字节串或字串	(134)
SHR	逻辑右移	(135)
STC	置进位标志	(137)
STD	置方向标志	(137)
STI	置中断标志	(137)
STOS/STOSB/STOSW	存贮字节串或字串	(138)
SUB	减法	(139)
TEST	测试	(140)
WAIT	等待	(142)
XCHG	交换	(142)
XLAT	转换	(143)
XOR	异一或	(144)
附录A. 错误信息		(146)
汇编程序的错误信息		(146)
Small 汇编程序 (ASM)		(146)
MACRO 汇编程序 (MASM)		(146)
I/O 处理程序的错误信息		(151)
运行期间的错误信息		(153)
附录B. 指令表		(153)
附录C. 伪操作表		(168)
数据伪操作		(168)
条件伪操作		(169)
宏伪操作		(170)
例表伪操作		(171)
附录D. 会话实例		(172)
要求		(172)
复制汇编程序的主软盘		(172)
建立用于汇编程序会话的源文件		(173)
汇编源程序		(173)
查看由汇编程序建立的文件		(174)
建立交叉引用表		(174)
其它模块的处理		(175)
连接目标文件产生可执行的应用程序		(176)
运行最后得到的应用程序		(177)
关于调试的几点注意事项		(177)
汇编程序与连接程序的清单		(179)
词汇		(191)

第一章 绪 论

IBM 多能 5550 MACRO 汇编程序把按照汇编语言编写的源代码转换为计算机识别的机器语言的目标模块。

Small 汇编程序——ASM (64k) 和 MACRO 汇编程序——MASM (96k) 可以任选使用。但最好使用 MACRO 汇编程序——MASM, 因为它支持本手册所叙述的全部功能。Small 汇编程序不支持 MACRO 以及和 MACRO 相关的功能, 即 REPT、IRP、IRPC, 还有 STRUC 和 RECORD 伪操作。

MARCO 汇编程序提供下列功能:

- 检测和编制源程序。
- 生成 MACRO 指令。
- 从原来分配的地址到其他地址再定位目标程序。
- 检测源程序中的错误。
- 为每个汇编的源程序产生源程序语句和目标程序语句的清单。

源程序在汇编处理期间并不执行, 只不过转换成机器语言而已。

MACRO 汇编程序是在 IBM 多功能 5550 磁盘操作系统 (DOS) 手册中叙述的磁盘操作系统 (DOS) 支持下运行的。

汇编语言是一种符号语言。它的每个操作代码都被写成易于记忆的符号字母, 这些符号字母通常称为记忆码。汇编语言在形式和内容上类似于机器语言, 它由表示指令和注释的语句组成。指令语句是语言的工作部分, 而且还可分为如下几个组:

- 机器指令
- 汇编程序指令
- 宏指令
- 伪操作

机 器 指 令

机器指令是汇编程序指令集中硬件指令的符号表示。

汇 编 程 序 指 令

汇编程序指令是在源模块汇编期间请求汇编程序执行的某些操作。

宏 指 令

宏指令是请求汇编程序处理预先规定的代码序列。按着这个序列，汇编程序生成源指令。然后把这些指令当做源模块中的初始输入部分处理。

伪 操 作

伪操作是与汇编程序指令的记忆码相似的记忆码。它告诉汇编程序使用数据、条件分支、宏指令以及列表来做什么。通常伪操作不产生机器语言代码（见第五章在 MACRO 汇编程序中使用的伪操作）。

EDLIN 程 序

EDLIN 程序可以建立、修改和显示输入给汇编程序的源文件。

EDLIN 程序是行文本编辑程序，它完成：

- 删除、编辑、插入和显示行。
- 检索、删除、替换和显示文本。
- 建立新的文件并进行保护。
- 更新老文件并保护更新后的文件和原文件。

有关 EDLIN 程序的详细叙述见 IBM 多功能 555D 磁盘操作系统手册。

处理器的结构

做为 MACRO 汇编程序的程序员，他应当了解 IBM 多功能 555D 处理器的结构，包括：

- 寻址
- 寄存器
- 栈

编 址

IBM 多功能 5550 的处理器把内存分为段，每个段包含 64k 字节的范围，而编址就是借助于每个段中的16个二进制位的位移来进行的。段值由一个16个字节的单位（称为节）来计算，并存储于四个具有16个二进制位的段寄存器中（CS、DS、ES、SS）。为了存取内存单元，处理器自动地把段地址的四个二进制位向左移动并加上存储于另外寄存器中的位移，形成一个二十个二进制位的物理地址。

寄 存 器

IBM 多功能 555D 的处理器使用下列寄存器：

- 通用寄存器 (Ax、Bx、Cx、Dx) 它们通过机器语言指令的控制而用于不同的操作。专用寄存器用于确定的用途（更详细的内容请参见第六章）。它与其他寄存器的区别是：通用寄存器可分成能够独立使用的高位 (H) 寄存器和低位 (L) 寄存器。
- 指针寄存器和变址寄存器 (SP、BP、SI、DI)。通常，这些寄存器用来存贮基地址或段中的位移。栈指针寄存器给出段中的位移，而变址寄存器给出数据段中的位移。
- 段寄存器 (CS、DS、SS、ES)，段寄存器存贮段地址。
- 指令指针 (IP)

指令指针总是存贮下一个要执行指令的地址。转移和调用通过使用特殊值替换指令指针中的内容来实现。指令指针的内容可以直接存取。

- (标志寄存器 (AF、CF、DF、IF、OF、PF、SF、TF、ZF))

标志寄存器表示当前处理条件的状态并控制处理器的功能。每个标志寄存器由十六位的标志寄存器字中的一位组成（关于每个标志的叙述可在第四章的“标志寄存器”一节中查到）。

栈

栈是存放返回地址和其他暂存值的内存区域。它的段地址总是存放在栈段寄存器 (SS) 中。指针寄存器的其中之一，即栈指针 (SP，总是指向栈顶) 定位在高端地址并且以每次在栈中存放 (进栈) 一个项来减小。换句话说，栈是朝着低地址增大的。

相反，若一个项从栈中取消 (退栈)，那么栈指针的值增大。为了在栈中存放或取消一项可使用指令 PUSH、POP、CALL 及 RET (请参阅第六章)。

第二章 MACRO 汇编程序

通过使用 IBM 多功能 5550 磁盘操作系统手册中叙述的 EDLIN 程序编写了汇编源程序后,就可对其进行汇编、连接、运行和调试。

如果汇编程序发现有错误,则把错误显示在清单文件上。用户应用编辑程序校正这些错误,然后对源程序重新进行汇编,直到产生无错误的机器语言为止。

在 IBM 多功能 5550 磁盘操作系统手册中描述的连接程序可以把汇编程序的机器语言(包括来自其他汇编运行的机器语言)转换为能够运行的程序。

最后产生的程序便可随时用于装入、运行和调试。

命 令 格 式

使用这本手册中的命令、语句、函数和变量、一定要遵循下列规则:

1. 用大写字母表示的字是关键字。它们必须按照所给出的形式使用,而且用大写字母或小写字母输入均可。
2. 必须给出小写字母表示的项。
3. 方括号 ([]) 中的项是任选项。
4. 省略号 (...) 表示在它前面的项可以重复出现多次。
6. 除方括号以外的所有标点,诸如逗号、圆括号,尖括号、斜线或分号必须在指定的位置上出现。

隐含的文件名扩展

隐含的文件名扩展由一个句点和三个字符 (.XXX) 组成。汇编程序使用的扩展如下所示。

扩 展	定 义
.ASM	汇编程序源文件的文件名扩展。
.CRF	表示由 CREF 实用程序使用的交叉引用文件的文件名扩展。
.LST	可打印的汇编程序清单文件的文件名扩展。
.OBJ	可再定位的目标文件的文件名扩展。
.REF	带有由 CREF 实用程序产生的交叉引用信息的可打印文件的文件名扩展。

怎样启动汇编程序

根据系统的配置, 可通过下面三个选择来启动 MACRO 汇编程序——MASM 或 Small 汇编程序——ASM。

- 当系统只有一个软盘驱动器时, 选择1允许在使用汇编程序命令之前, 更换软盘。
- 如果系统有一个以上的软盘驱动器, 那么选择 2 允许直接使用汇编程序命令。
- 选择 3 允许使用由磁盘操作系统 (DOS) 提供的批命令功能。这个功能提供了组合汇编程序命令的一种手段。它能自动地启动汇编程序。

如何用选择 1 启动汇编程序

键盘输入:

ASM

或

MASM

注意: 使用(M)ASM 表示选择哪一种汇编程序。

系统从软盘上加载汇编程序, 稍隔片刻, 汇编程序给出下面提示:

Source filename [.ASM]:

在响应这个提示之前, 可用装有源文件的软盘替换汇编程序软盘。

注意:

1. 若不给出文件名扩展, 则汇编程序将把括号中的名字作为隐含的文件名扩展使用。
2. 虽然在不给出文件名扩展的情况下, 汇编程序提供隐含的文件名扩展, 但也可以显式指定带有新扩展的文件名代替所有其他的扩展。
3. 隐含的软盘驱动器为 DOS 使用的隐含驱动器。当然, 也可以通过把驱动器 ID 作为文件说明中的一部分来显式代替。

Source filename 是已被存贮的源程序文件的名称。如下所示, 假定用“myfile”响应了这个提示:

Source filename [.ASM]: myfile

汇编程序能够自动地查寻 .ASM 文件名, 所以不必给出 .ASM 文件名扩展。给出源文件名后, 在屏幕上可看到下面的提示:

Object filename (MYFILE. OBJ):

Object filename 是必须有的目标 (计算机可识别的) 文件的名称。如果以 MYFILE.OBJ 作为目标文件的名称, 那么只按 Enter 键。否则可以给出其他的文件名, 在给出的文件名的后面汇编程序将加上文件名扩展 .OBJ (.OBJ 可以被其他的扩展显