



東北大學
Northeastern University

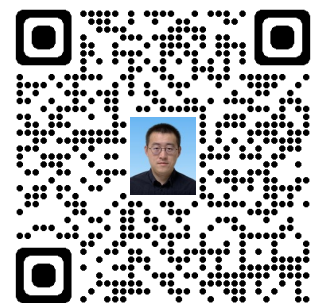
汇编语言程序设计

主讲：刘松冉

单位：东北大学计算机学院
智慧系统国际联合实验室

联系方式：liusongran@cse.neu.edu.cn

个人主页：<http://faculty.neu.edu.cn/liusongran>
<https://liusongran.github.io/>



第九章 条件汇编与宏指令

一. 条件汇编伪操作

二. 宏伪操作

- 条件汇编伪操作和宏伪操作命令也是伪指令的一部分
- 具有条件汇编宏伪操作功能的汇编程序称为宏汇编程序
- Microsoft的MASM就是宏 (MACRO)汇编程序,它提供了丰富的条件汇编伪操作和宏伪操作命令。

一. 条件汇编伪操作

▶ 条件汇编伪操作允许用户在编制汇编语言程序时规定某种条件，汇编程序在汇编过程中测试条件是否成立，当条件成立时，则将某段汇编语言源程序汇编为目标程序，否则，不进行汇编。

Microsoft宏汇编程序提供了十种条件汇编伪操作命令，IF、IF1、IF2、IFE、IFDEF、IFNDEF、IFB、IFNB、IFIDN、IFDIF。

格式：IFx [cont]

：若干语句(称为条件块)

[ELSE] (可选的)

：若干语句(称为条件块)

ENDIF

一. 条件汇编伪操作

- ▶ 1) **IF** CONT 当条件表达式CONT值不是零时，条件为真。
- 2) **IFE** CONT 当条件表达式CONT值为零时，条件为真。
- 3) **IF1** 如果是第一遍扫描，条件为真。
- 4) **IF2** 如果是第二遍扫描，条件为真。
- 5) **IFDEF** CONT 此操作中的条件CONT是一个符号，如果这个符号已经定义，或者已用EXTRN伪指令说明，则条件为真。
- 6) **IFNDEF** CONT 该伪操作中的条件CONT是一个符号，如果此符号未定义或未经EXTRN伪指令说明，则条件为真。
- 7) **IFB** <CONT> 如果尖括号中为空，则条件为真。
- 8) **IFNB** <CONT> 如果尖括号中不为空，则条件为真。
- 9) **IFIDN** <CHR1>,<CHR2> CHR1和CHR2为字符串，如果两个字符串相同，则条件为真。
- 10) **IFDIF** <CHR1>,<CHR2> CHR1和CHR2为字符串，如果两个字符串不相同，则条件为真。

一. 条件汇编伪操作

```
ARG1 EQU 35H
ARG2 = NOT ARG1
IF (ARG1 OR ARG2) EQ 0FFFFH
MOV AX, ARG1
MOV BX, ARG2
ADD AX, BX
IF (ARG1 AND ARG2) EQ 0FFFFH
SUB AX, CX
IFE ARG1
ADD AX, DX
ENDIF
MOV [SI], AX
ENDIF
MOV [DI], AX
ENDIF
```

第九章 条件汇编与宏指令

一. 条件汇编伪操作

二. 宏伪操作

- 1) 宏定义与宏结束伪操作命令
- 2) 参数的使用
- 3) 宏中的标号处理
- 4) 宏嵌套
- 5) 宏与子程序的区别

- 宏伪操作命令可以把多次使用的功能程序定义为一个名字，汇编程序（汇编器）在对源程序进行汇编时，就将它所代表的的源程序的机器代码写在宏名字出现处。
- 也就是说已经定义的宏名字可以作为指令一样使用，为与机器指令相区别，称之为宏指令。

二. 宏伪操作

1). 宏定义的一般格式：

MNAME MACRO [DUMPAR1],[DUMPAR2].....

:

:

:

ENDM

宏体

说明：

- 1) MNAME：宏名字
- 2) MACRO：宏定义伪指令；
- 3) ENDM：宏结束伪指令；
- 4) DUMPAR1..：形式参数，也称为哑参数，可选项。形式参数在宏扩展时被实际的参数替代。

```
2  BLMOV  MACRO  SRC, DST, CNT    ;宏定义
3
4      LEA  SI, SRC
5      LEA  DI, DST
6      MOV  CX, CNT
7      CLD
7      REP  MOVSB
8      ENDM                        ;宏结束
```

二. 宏伪操作

▶ 2). 参数的使用（分类）：

- 用参数代表自定义符号（地址）和数值（例，BLMOV ADR1, ADR2, CNT1）
- 用参数代表指令
- 用参数代替寄存器名字
- 用参数代表任何一字符串或子字符串

2	BLMOV	MACRO	SRC, DST, CNT	;宏定义
3		LEA	SI, SRC	
4		LEA	DI, DST	
5		MOV	CX, CNT	
6		CLD		
7		REP	MOVSB	
8		ENDM		;宏结束

二. 宏伪操作

▶ 2). 参数的使用（分类）：

- 用参数代表自定义符号（地址）和数值
- 用参数代表指令（例，CLEAR CLD, ADR1, 80）
- 用参数代替寄存器名字
- 用参数代表任何一字符串或子字符串

10	CLEAR	MACRO	DIR, DST, CNT
11		DIR	
12		LEA	DI, DST
13		MOV	CX, CNT
14		XOR	AL, AL
15		REP	STOSB
16		ENDM	

二. 宏伪操作

▶ 2). 参数的使用（分类）：

- 用参数代表自定义符号（地址）和数值
- 用参数代表指令
- 用参数代替寄存器名字（例，RLS R, AX, 5 或 RLS L, CH, 2）
- 用参数代表任何一字符串或子字符串

18	RLS	MACRO	DIR, REG, CNT
19		MOV	CL, CNT
20		R0&DIR	REG, CL
21		ENDM	

二. 宏伪操作

▶ 2). 参数的使用（分类）：

- 用参数代表自定义符号（地址）和数值
- 用参数代表指令
- 用参数代替寄存器名字
- 用参数代表任何一字符串或子字符串
(例，MADD1 LOOP1, ADR3, ADR1, 8, ADR2))

```
23 ;定义两个多精度数据求和的宏指令
24 MADD1 MACRO LAB, SRC, DST, LEN, SUN
25     LEA SI, SRC
26     LEA DI, DST
27     MOV DX, LEN
28     LEA BX, SUN
29
30 LAB: MOV AL, [SI]
31     ADC AL, [DI]
32     MOV [BX], AL
33     INC SI
34     INC DI
35     INC BX
36     LOOP LAB
37 ENDM
```

二. 宏伪操作

▶ 3). 宏中的标号处理

编写汇编语言程序时的一个规则是标号必须是唯一的，即同一个标号不允许在一个程序的标号域中出现两次以上，否则为重复定义，汇编器会在汇编过程中打印错误信息。解决这个问题有两种方法：

- 把标号定义为参数形式
- 在宏定义中声明标号为局部标号

```
23 ;定义两个多精度数据求和的宏指令
24 MADD1 MACRO LAB, SRC, DST, LEN, SUN
25     LEA SI, SRC
26     LEA DI, DST
27     MOV DX, LEN
28     LEA BX, SUN
29     CLC
30 LAB: MOV AL, [SI]
31     ADC AL, [DI]
32     MOV [BX], AL
33     INC SI
34     INC DI
35     INC BX
36     LOOP LAB
37     ENDM
```

二. 宏伪操作

▶ 3). 宏中的标号处理

编写汇编语言程序时的一个规则是标号必须是唯一的，即同一个标号不允许在一个程序的标号域中出现两次以上，否则为重复定义，汇编器会在汇编过程中打印错误信息。解决这个问题有两种方法：

- 把标号定义为参数形式
- 在宏定义中声明标号为局部标号

```
39      ;定义求三个寄存器中最小数的宏指令
40      FMIN      MACRO      REG1, REG2, REG3
41          LOCAL  NEXT1, NEXT2
42          CMP     REG1, REG2
43          JNA     NEXT1
44          MOV     REG1, REG2
45      NEXT1:     CMP     REG1, REG3
46          JNA     NEXT2
47          MOV     REG1, REG3
48      NEXT2:     NOP
49          ENDM
```

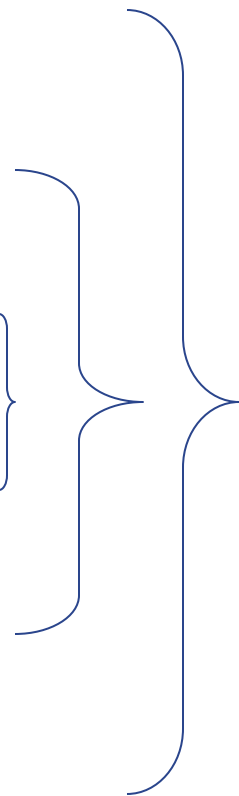
二. 宏伪操作

▶ 4). 宏嵌套

宏嵌套有两种形式：宏定义中嵌套宏定义和宏定义中嵌套宏调用，两种宏嵌套的深度不限：

- 宏定义中嵌套宏定义

```
MAC1  MACRO
...
MAC2  MACRO
...
MAC3  MACRO
...
      ENDM
      ...
      ENDM
      ...
      ENDM
```



当宏定义中嵌套宏定义时，必须首先调用最外层宏定义，然后才能调用内层宏定义。下面是一个宏嵌套的例子。

二. 宏伪操作

▶ 4). 宏嵌套 – 宏定义中嵌套宏定义

BHTOA1	MACRO	
	MOV	AH, AL
AHHN	MACRO	
	LOACAL	AHHN1
	MOV	CL, 4
	SHR	AH, CL
	CMP	AH, 10
	JC	AHHN1
	ADD	AH, 7
AHHN1 :	ADD	AH, 30H
	ENDM	
ALLN	MACRO	
	LOCAL	ALLN1
	AND	AL, 0FH
	CMP	AL, 10
	JC	ALLN1
	ADD	AL, 7
ALLN1 :	ADD	AL, 30H
	ENDM	
	ENDM	

则宏扩展后得到如下程序段：

```

      BHTOA1
1      MOV      AH,AL
      AHHN
1      MOV      CL,4
1      SHR      AH,CL
1      CMP      AH,10
1      JC       ??0000
1      ADD      AH,7
1      ??0000:  ADD      AH,30H
      ALLN
1      AND      AL,0FH
1      CMP      AL,10
1      JC       ??0001
1      ADD      AL,7
1      ??0001:  ADD      AL,30H
```

当宏定义中嵌套宏定义时，必须首先调用最外层宏定义，然后才能调用内层宏定义。下面是一个宏嵌套的例子。有如下宏调用：

- BHTOA1
- AHHN
- ALLN

二. 宏伪操作

4). 宏嵌套

宏嵌套有两种形式：宏定义中嵌套宏定义和宏定义中嵌套宏调用，两种宏嵌套的深度不限：

- 宏定义中嵌套宏调用

MACA **MACRO**

...

MACB

...

ENDM

MACB **MACRO**

...

ENDM

各宏定义可单独调用。

```
51  BHT0A2: MACRO
52          PUSH    CX
53          MOV     CH, AL
54          MOV     CL, 04
55          SHR     AL, CL
56          HTOA          ;宏调用
57          MOV     AH, AL
58          MOV     AL, CH
59          AND     AL, 0FH
60          HTOA          ;宏调用
61          POP     CX
62          ENDM
63  HTOA    MACRO
64          LOCAL   HTOA1
65          AND     AL, 0FH
66          CMP     AL, 10
67          JC      HTOA1
68          ADD     AL, 07
69  HTOA1:  ADD     AL, 20H
70          ENDM
```

```
72  ;宏调用
73  ...
74  MOV     AL, 05
75  HTOA
76  ...
77  MOV     AL, 47H
78  BHT0A2
79  ...
```


二. 宏伪操作

▶ 5). 宏与子程序的区别

	宏	子程序
处理者	宏汇编程序	中央处理器
处理时间	在汇编过程中	程序执行时
处理方法	用宏体替换宏指令	用CALL指令调用
占用内存	占用内存空间多	占用内存空间少
运行速度	运行速度较高	运行速度较低
灵活性	十分灵活	不够灵活