

8086寻址方式

8086/8088有七种基本的寻址方式：立即寻址，寄存器寻址，直接寻址，寄存器间接寻址，寄存器相对寻址，基址变址寻址，相对基址变址寻址。

8086的机器码格式为： `操作码` | `寻址方式` | `位移量` | `立即数`

立即数寻址

立即数：操作数包含在指令中，它作为指令的一部分，跟在操作码后存放在代码段。这种操作数称为立即数。

立即寻址方式所提供的操作数紧跟在操作码后面，与操作码一起放在指令代码段中，不需要到其他地址单元中去取。立即数可以是8位（imm8），也可以是16位（imm16）。这种寻址方式主要用于给寄存器或存储单元赋初值的场合。

```
MOV AX, 1234H ; 给AX寄存器赋值为1234H
```

这种寻址方式主要用于给寄存器或存储单元赋初值的场合。

寄存器寻址

寄存器寻址的特点是操作数在CPU内部的寄存器中，在指令中指定寄存器号。

对于16位操作数，寄存器可以是AX、BX、CX、DX、SI、DI和SP等；

对于8位操作数，寄存器可以是AL、AH、BL、BH、CL、CH、DL和DH。

```
MOV SI, AX
MOV AL, DH
```

直接寻址

直接寻址的操作数在存储器中，指令直接包含有操作数的有效地址。由于操作数一般存放在数据段，所以操作数的地址由DS加上指令中给出的16位偏移得到。

```
MOV AX, [5678H] ; 自动引用DS作为段寄存器
MOV AX, ES:[5678H] ; 指定ES作为段寄存器
```

这种寻址方式常用于处理单个存储器变量的情况。它可以实现在64K字节的段内寻找操作数。直接寻址的操作数通常是程序使用的变量。

寄存器间接寻址

寄存器间接寻址的操作数在存储器中，操作数有效地址在SI、DI、BX、BP这四个寄存器之一中。在一般情况下，如果有效地址在SI、DI和BX中，则以DS段寄存器的内容为段值；如果有效地址在BP中，则以SS段寄存器的内容为段值。

```
MOV AX, [SI] ; 自动引用DS作为段寄存器
MOV DL, CS:[BX] ; 引用的段寄存器是CS
MOV [BP], CX ; 此时引用的段寄存器是SS
```

这种寻址方式可以用于表格处理，在处理完表中的一项后，只要修改指针寄存器的内容就可以方便地处理表中的另一项。

寄存器相对寻址

操作数在存储器中，操作数的有效地址是一个基址寄存器（BX、BP）或变址寄存器的（SI、DI）内容加上指令中给定的8位或16位位移量之和。即 $EA = (BX) \text{ 或 } (BP) \text{ 或 } (SI) \text{ 或 } (DI) + 8 \text{ 位或 } 16 \text{ 位位移量}$

在一般情况下，如果SI、DI或BX的内容作为有效地址的一部分，那么引用的段寄存器是DS。

```
MOV AX, [DI+1223H] ; 引用的段寄存器是DS
```

而当BP的内容作为有效地址一部分时，那么引用的段寄存器是SS。

```
MOV BX, [BP-4] ; 引用的段寄存器是SS
```

也可以自行指定段寄存器。

```
MOV ES:[BX+5], AL ; 引用ES作为段寄存器
```

这种寻址方式同样可用于表格处理，表格的首地址可设置为指令中的位移量，利用修改基址或变址寄存器的内容来存取表格中的项值。

以下两种写法等价：

```
MOV AX, [SI+3]
MOV AX, 3[SI]
```

基址变址寻址

操作数在存储器中，操作数的有效地址由基址寄存器之一的内容与变址寄存器之一的内容相加得到。即 $EA = (BX) \text{ 或 } (BP) + (SI) \text{ 或 } (DI)$

在一般情况下，如果BP的内容作为有效地址的一部分，则以SS的内容作为段值，否则以DS的内容作为段值。

```
MOV AX, [BX+DI] ; 此时引用DS段寄存器
MOV AX, ES:[BX+SI] ; 引用ES作为段寄存器
MOV DS:[BP+SI], AL ; 引用DS作为段寄存器
```

这种寻址方式适用于数组或表格处理。用基址寄存器存放数组首地址，而用变址寄存器来定位数组中的各元素，或反之。

以下两种表示方法是等价的：

```
MOV AX, [BX+DI]
MOV AX, [DI][BX]
```

相对基址变址寻址

操作数在存储器中，操作数的有效地址由基址寄存器之一的内容与变址寄存器之一的内容及指令中给定的8位或16位位移量相加得到。即 $EA = (BX) \text{ 或 } (BP) + (SI) \text{ 或 } (DI) + \text{8位或16位位移量}$

在一般情况下，如果BP作为有效地址的一部分，则以SS段寄存器的内容为段值，否则以DS段寄存器的内容为段值。

在指令中给定的8位或16位位移量采用补码形式表示。在计算有效地址时，如果偏移量是8位，那么被带符号扩展成16位。当所得的有效地址超过FFFFH时，就取其64K的模。

```
MOV AX, [BX+DI-2] ; 自动引用DS作为段寄存器
```

以下四种表示方法均是等价的：

```
MOV AX, [BX+DI+1234H]
MOV AX, 1234H[BX+DI]
MOV AX, 1234H[BX][DI]
MOV AX, 1234H[DI][BX]
```

总结

寻址方式	指令特点	例子	注意
立即寻址	用立即数给寄存器赋值	mov ax, 1234h	无
寄存器寻址	源操作数为寄存器	mov ax, bx	无
直接寻址	括号内一个立即数	(1) mov dx, [1000h] (2) mov es: [1000h], dx	使用 <code>ds</code> 作为缺省段寄存器
寄存器间接寻址	括号内一个基址或变址寄存器(<code>SI</code> , <code>DI</code> , <code>BX</code> , <code>BP</code>)	(1) mov dx, [di] (2) mov [bp], cx	若有效地址在 <code>SI</code> , <code>DI</code> 和 <code>BX</code> 中, 则以 <code>ds</code> 作为缺省段寄存器; 若有效地址在 <code>BP</code> 中, 则以 <code>ss</code> 作为缺省段寄存器

寻址方式	指令特点	例子	注意
寄存器相对寻址	括号内一个基址或变址寄存器(SI, DI, BX, BP)+ 一个立即数	(1) mov ax, [bx+16] (2) mov ax, 16[bx] (3) mov es: [bx+5], al	若 SI, DI 和 BX 作为有效地址的一部分, 则以 DS 作为缺省段寄存器; 若 BP 作为有效地址的一部分, 则以 SS 作为缺省段寄存器
基址变址寻址	括号内一个基址寄存器(BX, BP)+ 一个变址寄存器(SI, DI)	(1) mov ax, [bx+si] (2) mov ax, [si][bx] (3) mov ds: [bp+si], al	若 SI, DI 和 BX 作为有效地址的一部分, 则以 DS 作为缺省段寄存器; 若 BP 作为有效地址的一部分, 则以 SS 作为缺省段寄存器
相对基址变址寻址	括号内一个基址寄存器(BX, BP)+ 一个变址寄存器(SI, DI)+ 一个立即数	(1) mov ax, [bx+di+10h] (2) mov ax, 10h[bx+di] (3) mov ax, 10h[bx][di] (4) mov ax, 10h[di][bx]	若 SI, DI 和 BX 作为有效地址的一部分, 则以 DS 作为缺省段寄存器; 若 BP 作为有效地址的一部分, 则以 SS 作为缺省段寄存器