

汇编快速入门(5)

By: 凌晨五点

第五节：指令及寻址方式

知识点

- ▣ 指令简介
- ▣ 七种寻址方式

指令简介

指令(1)

- ▣ 一般来说指令是由两部分组成，即操作码和操作数。



在指令中操作码是不可缺少的，但操作数可以没有，也可以有一个操作数或两个操作数

指令(2)

根据操作数的个数，指令格式可分为以下几种：

- ▣ 1. 零操作数指令

指令格式中没有操作数或操作数是**隐含约定**的。

- ▣ 2. 一操作数指令

指令格式中有一个操作数，或还有一个隐含的操作数（实际上是双操作数）。

- ▣ 3. 二操作数指令

指令中有两个操作数，其中一个为**目的操作数**，另一个为**源操作数**。

指令(3)

由此可见，操作数可分为**源操作数**和**目的操作数**。

- ▣ 源操作数：只能读取的操作数。
- ▣ 目的操作数：即可读取又可写入（存放操作结果）的操作数



寻址方式

寻址方式

- ▣ **寻址方式**：指令中指明**操作数**存放位置的表达方式

指令中数据存放的位置（1）

指令中进行操作数的数据存放位置有三种情况：

▣ 1) 存放指令中（立即数）

- 操作数包含在指令中，即被操作数据直接表示在指令的操作数字段中，也就是说紧跟在操作码之后。

- 例如：

- ▣ MOV AL, 10H

- ▣ 这种操作数据称为立即数

▣ 2) 存于寄存器中（寄存器操作数）

- 数据存放在CPU的一个寄存器中。

- 例如：

- ▣ INC CX

指令中数据存放的位置（2）

▣ 3) 存于存储器中（**存储器操作数**）

- 数据在内存中或在I/O端口中，存放数据的**偏移地址**以某种方式表示在指令中。

- 例如：

- ▣ `MOV AX, [2500H]`

其中[2500]为存储器操作数

存储器操作数中操作数字段指示此操作数的偏移地址，而段地址由某个段寄存器来提供。此例中默认为数据段DS

寻址方式分类

- ▣ 寻址方式可以分为:
 - 立即数寻址方式
 - 寄存器寻址方式
 - 存储器寻址方式
 - ▣ 直接寻址方式
 - ▣ 寄存器间接寻址方式
 - ▣ 寄存器相对寻址方式
 - ▣ 基址加变址寻址方式
 - ▣ 相对加基址变址寻址方式

1.立即数寻址 (1)

- ▣ **立即数寻址**：操作数为立即数，直接存放在指令的操作数字段中
- ▣ 立即数寻址时，只允许**源操作数为立即数**，**目标操作数必须是寄存器或存储器**，其作用是给寄存器或存储单元赋值。
- ▣ 在汇编中，立即数不能作为指令中的第一操作数。

例如：MOV AL, 80H
MOV AX, 1234H

1.立即数寻址 (2)

MOV AL, 80H

AL

80H

程序存储器

操作码字节

80H

指令码

MOV AX, 1234H

AH

12H

AL

34H

程序存储器

操作码字节

34H

12H

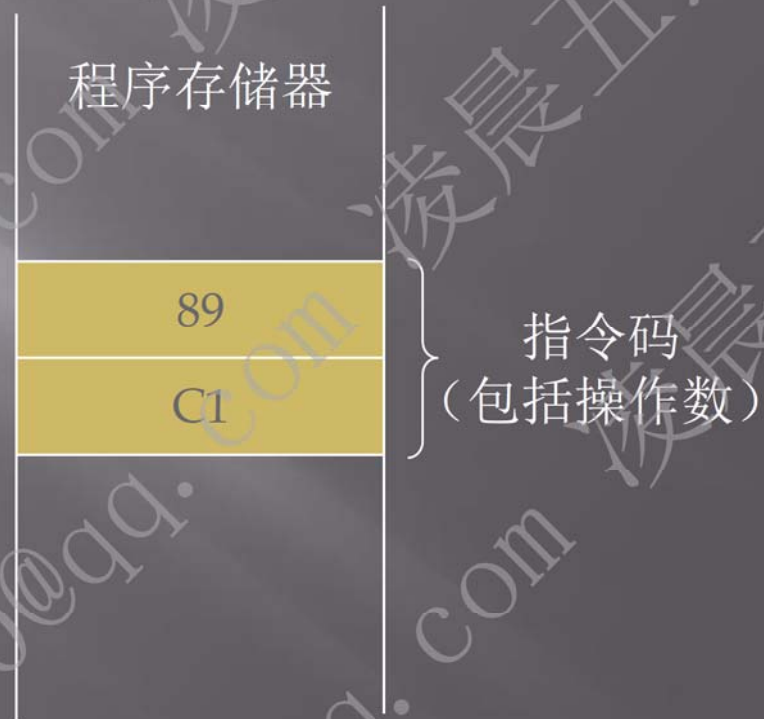
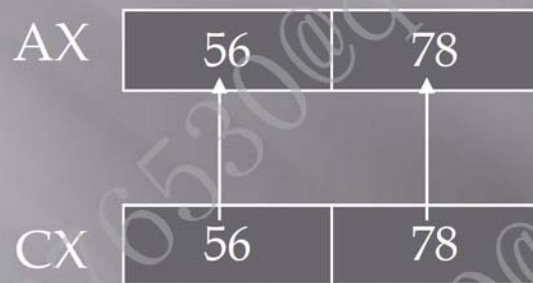
指令码

2.寄存器寻址（1）

- ▣ **寄存器寻址**：操作数在指令所指示的寄存器中
- ▣ 表示格式：直接在指令中写出寄存器名称

例如：INC BX
MOV AX, CX

2.寄存器寻址 (2)



3. 存储器寻址

- ▣ **存储器寻址**：操作数在指令所指示的寄存器中
- ▣ 表示格式：直接在指令中写出寄存器名称
- ▣ 包括如下方式：
 - 直接寻址方式
 - 寄存器间接寻址方式
 - 寄存器相对寻址方式
 - 基址加变址寻址方式
 - 相对基址加变址寻址方式

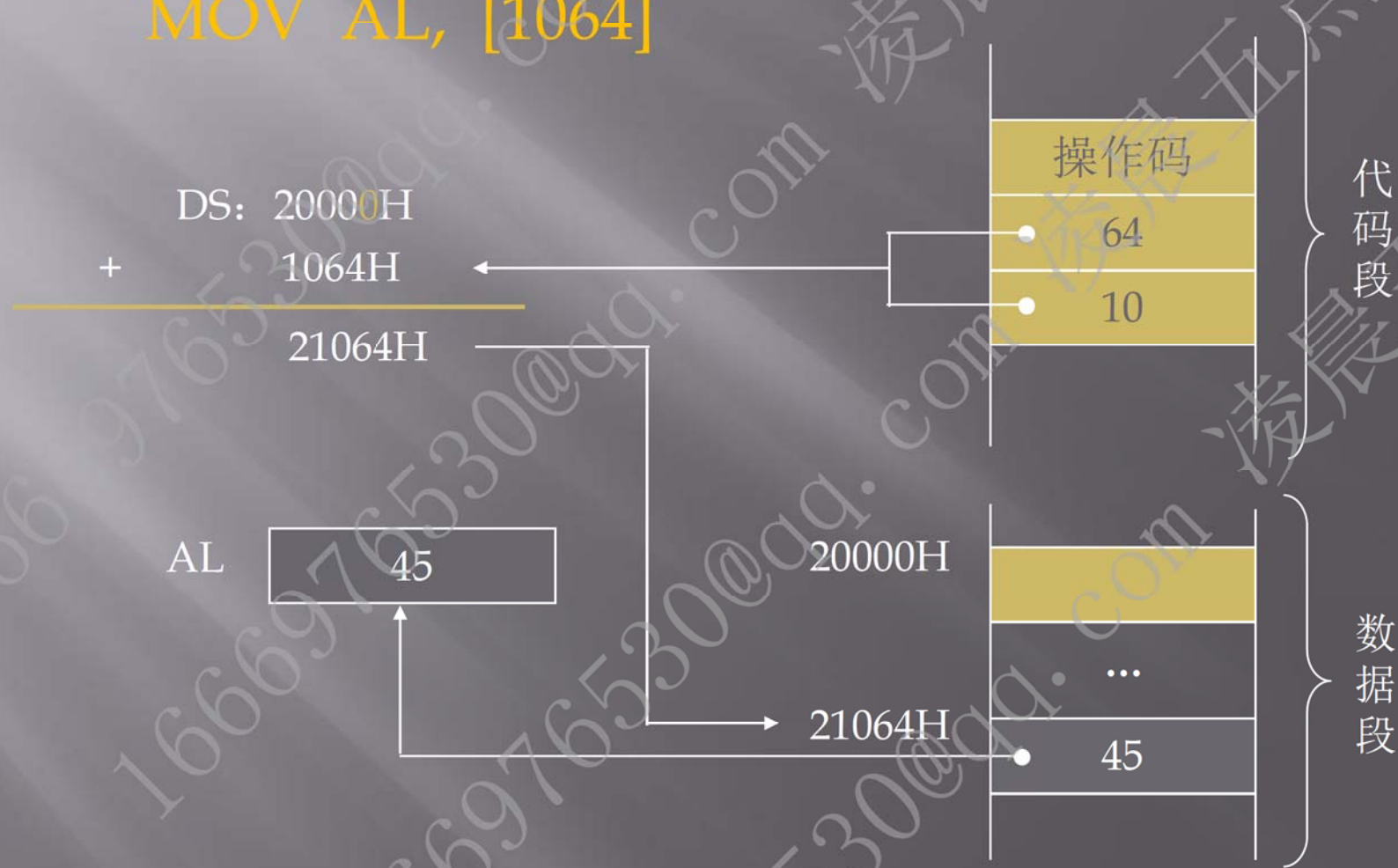
3.1.直接寻址（1）

- ▣ **直接寻址**：操作数存在内存中，操作数的**偏移地址**直接表示在指令中
- ▣ 表示格式：[偏移地址]
- ▣ **默认操作数**存放在内存的**数据段**中

例如：MOV AL, [1064H]

3.1.直接寻址 (2)

MOV AL, [1064]



段超越

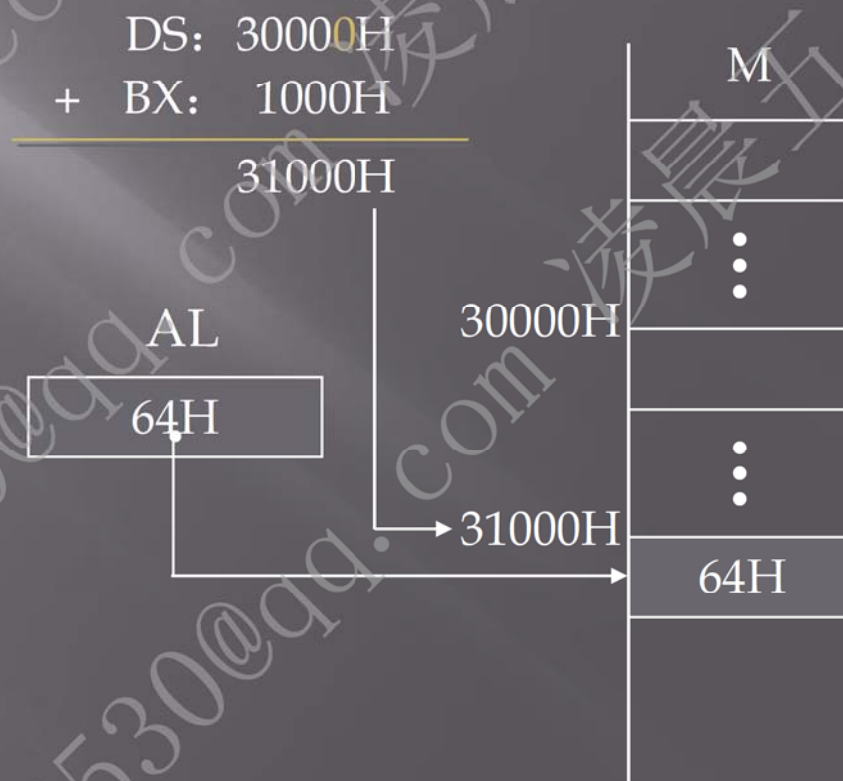
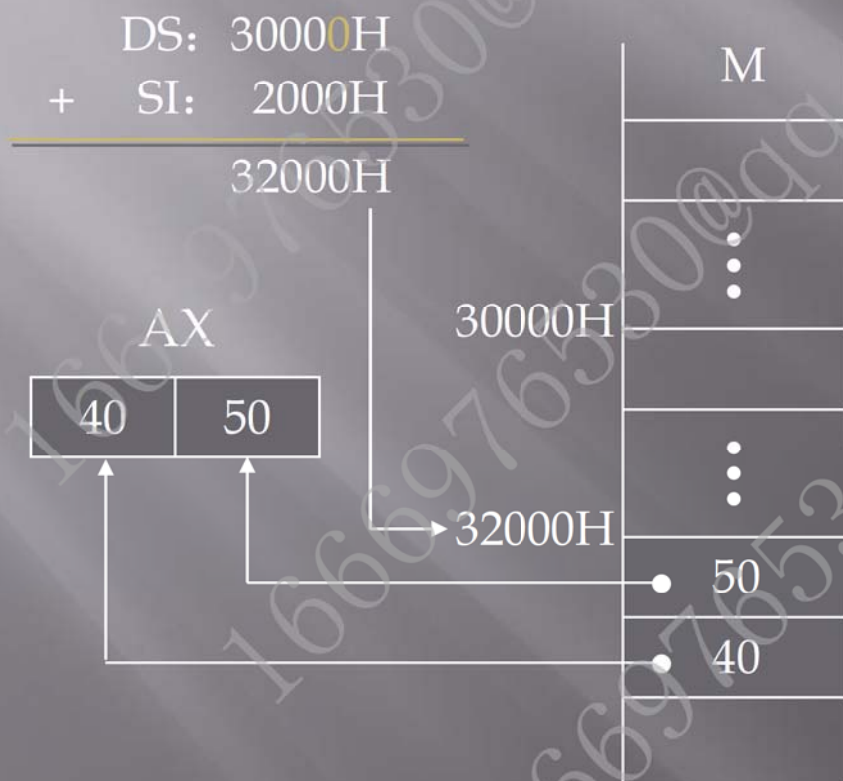
- ▣ 操作数也允许存放在其他段中（ES，SS），此时应该在指令中指明**段超越**
- ▣ **段超越**
 - 若操作数不在指令默认的段中，而在其它某个段中，则需要在指令中加以表示，这种情况称为段超越
- ▣ 例如
 - 直接寻址方式中操作数在附加段中，则应表示为：
MOV AL, ES:[1064H]

3.2.寄存器间接寻址方式 (1)

- ▣ 操作数存在存储器中，操作数的**偏移地址**在BX、SI、DI和BP的某个寄存器中。
- ▣ 若以**BX、SI、DI**作为间接寻址寄存器
则**默认操作数**存放在**数据段**中，用**DS**寄存器中的内容作为地址。
- ▣ 若以**BP**作为间接寻址寄存器
则**默认操作数**存放在**堆栈段**中，用**SS**寄存器中的内容作为地址。

3.2.寄存器间接寻址方式 (2)

```
MOV AX, [SI]
MOV [BX], AL
```

$$EA =$$


3.2.寄存器间接寻址方式（3）

▣ 寄存器间接寻址方式也允许**段超越**。

▣ 如

MOV AX, **DS**: [BP]

MOV CH, **SS**: [SI]

▣ 若以**BP**作为间接寻址寄存器

则**默认操作数**存放在**堆栈段**中，用**SS**寄存器中的内容作为地址。

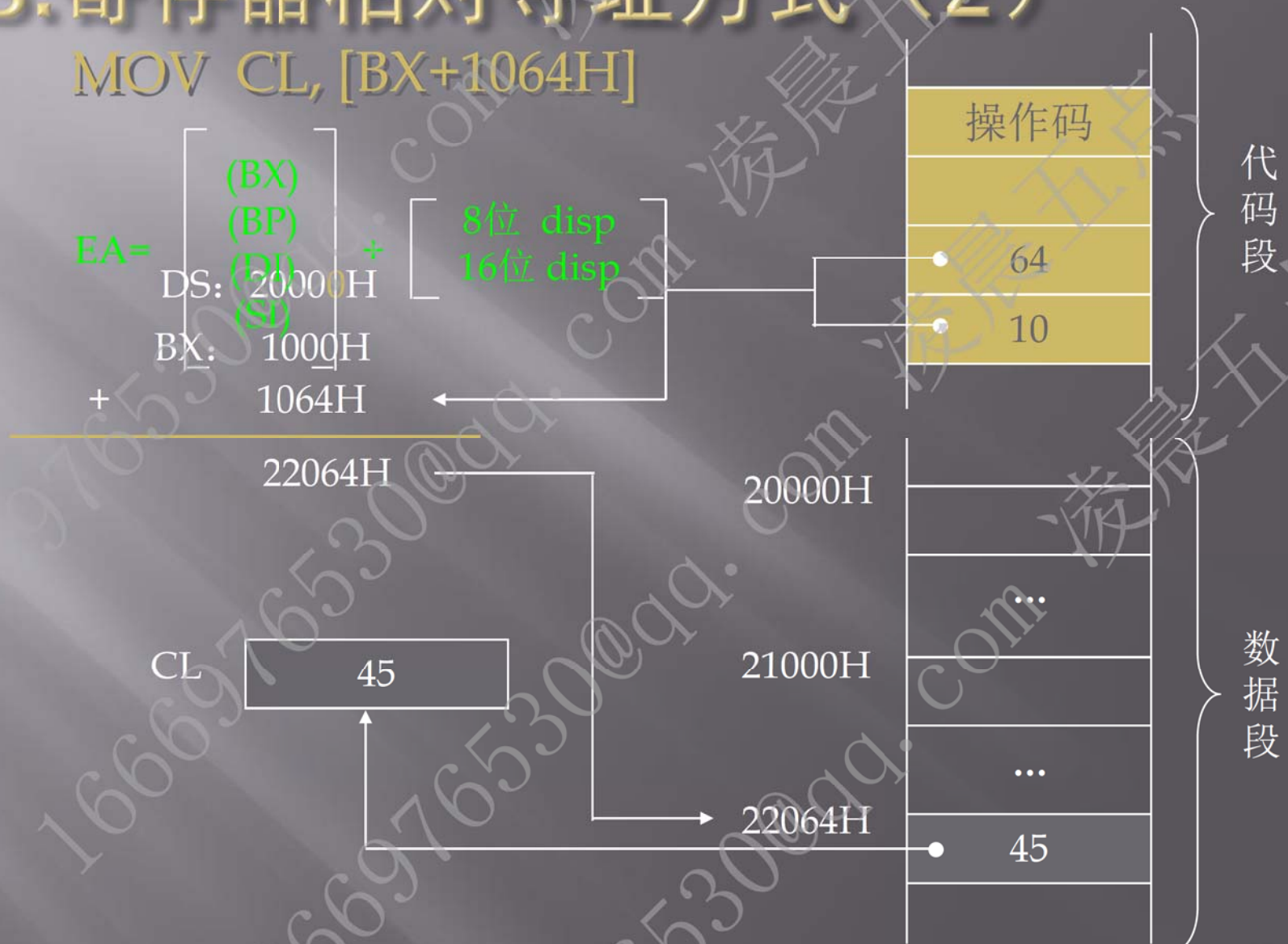
3.3.寄存器相对寻址方式 (1)

- ▣ 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(SI、DI)**的内容加上指令中给定的8位或16位位移量之和。
- ▣ 如果SI、DI、或BX中的内容作为有效地址的一部分，那么引用的段寄存器是DS；如果BP中的内容作为有效地址的一部分，那么引用的段寄存器是SS
 - 下面指令中，**源操作数采用寄存器相对寻址**，引用的段寄存器是SS: MOV BX,[BP-4]
 - 下面指令中，**目的操作数采用寄存器相对寻址**，引用的段寄存器是ES: MOV ES:[BX+5],AL

例如: MOV CL, [BX+1064H]

3.3.寄存器相对寻址方式 (2)

MOV CL, [BX+1064H]



3.4.基址加变址寻址方式 (1)

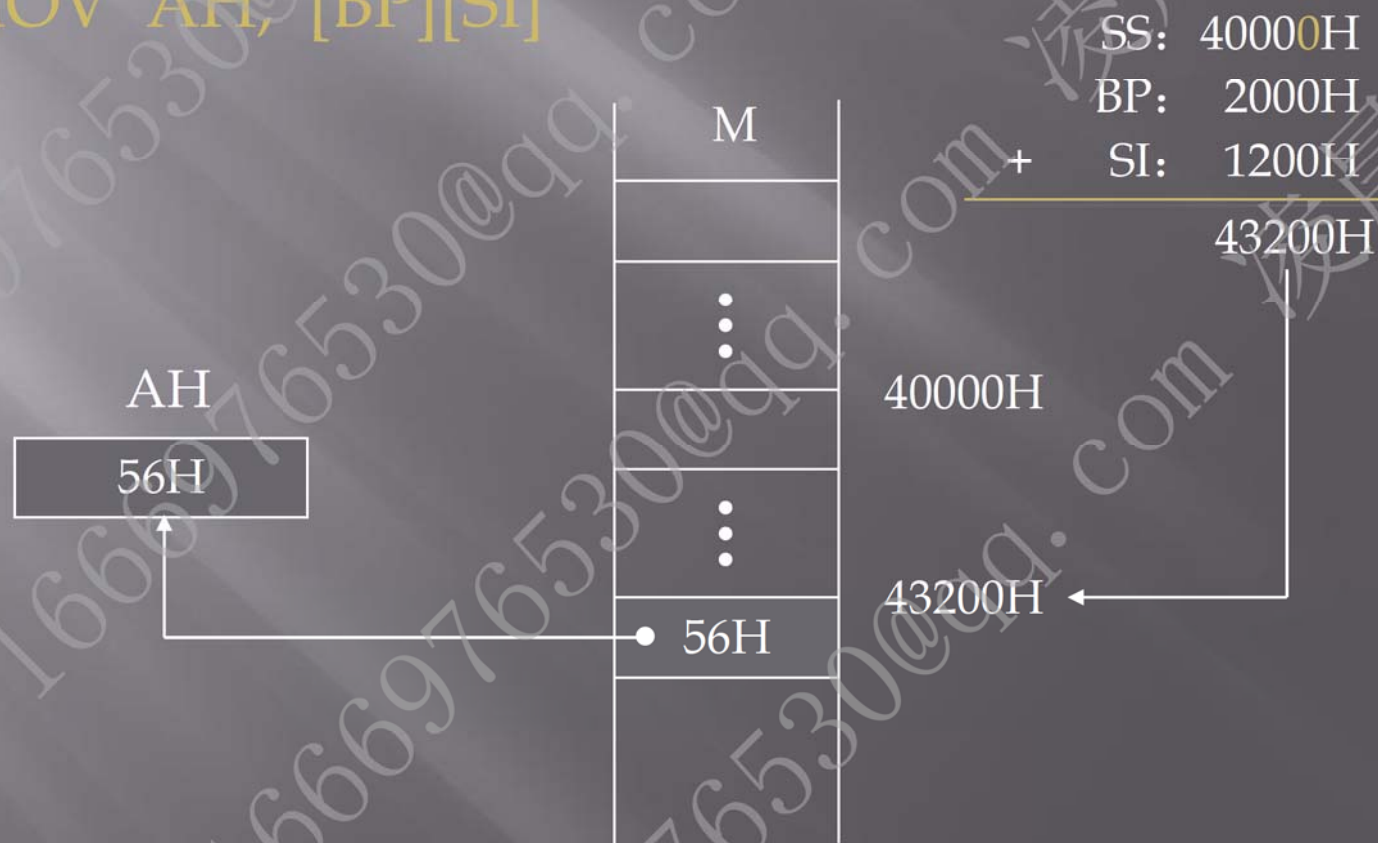
- ▣ 在基址加变址寻址方式中，通常把BX和BP看作是基址寄存器，把SI和DI看作变址寄存器，可把两种方式组合起来形成一种新的寻址方式。基址加变址的寻址方式是把一个基址寄存器BX或BP的内容，加上变址寄存器SI或DI的内容，并以一个段寄存器作为地址基准，作为操作数的地址。

例如：MOV AH, [BP][SI]

3.4.基址加变址寻址方式 (2)

$$EA = \begin{bmatrix} (BX) \\ (BP) \end{bmatrix} + \begin{bmatrix} (DI) \\ (SI) \end{bmatrix}$$

MOV AH, [BP][SI]



3.5.相对基址加变址寻址方式（1）

- ▣ 在相对基址变址寻址方式中，通常把BX和BP看作是基址寄存器，把SI和DI看作变址寄存器。它是把一个基址寄存器BX或BP的内容，加上变址寄存器SI或DI的内容，再加上指令中给定的8位或16位位移量，并以一个段寄存器作为地址基准，作为操作数的地址。
- ▣ 当基址寄存器为BX时，段寄存器使用DS。
- ▣ 当基址寄存器为BP时，段寄存器则用SS。

例如：MOV AL, [1064H]

3.5. 相对基址加变址寻址方式 (2)

$$EA = \begin{bmatrix} (BX) \\ (BP) \end{bmatrix} + \begin{bmatrix} (DI) \\ (SI) \end{bmatrix} + \begin{bmatrix} 8\text{位 disp} \\ 16\text{位 disp} \end{bmatrix}$$

MOV [BX+DI+1234H], AH

DS: 40000H

BX: 0200H

DI: 0010H

+

1234H

41444H

AH

45

40000H

41444H

操作码

34

12

...

...

45

代码段

数据段

总结



- ▣ 2. 寻址方式可以分为:
- 立即数寻址方式
 - 寄存器寻址方式
 - 存储器寻址方式
 - ▣ 直接寻址方式
 - ▣ 寄存器间接寻址方式
 - ▣ 寄存器相对寻址方式
 - ▣ 基址加变址寻址方式
 - ▣ 相对加基址变址寻址方式

谢谢



看雪公众号: [ikanxue](#)