

第3-1章 寻址方式

主讲：许万茹





本章教学内容

- 立即数寻址和寄存器寻址
- 存储器寻址
- 指令寻址

重点：操作数的各种寻址方式，指令寻址方式、段寄存器和偏移地址寄存器的关联



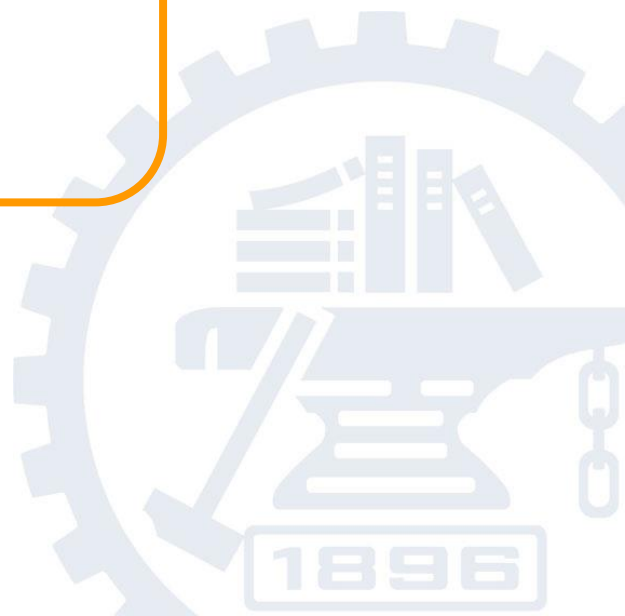
北京交通大学
BEIJING JIAOTONG UNIVERSITY



主要内容

3.1.1 操作数寻址

3.1.2 指令寻址





8086汇编指令格式

例：

MOV

AX , BX

指令 = 操作码 + 操作数

指示计算机执行什么操作

指明参加操作的数或者它所在的地址

将寄存器BX的数据传送到寄存器AX



寻址方式

- 为取得操作数或指令地址所使用的方法
 - **操作数寻址**：与数据有关的寻址方式
 - **指令寻址**：与转移地址有关的寻址方式

顺序寻址 跳转寻址





3.1.1 数据寻址方式

(1) 操作数包含在指令中——立即数寻址

(2) 操作数在CPU的内部寄存器中

——寄存器寻址

(3) 操作数在存储器中

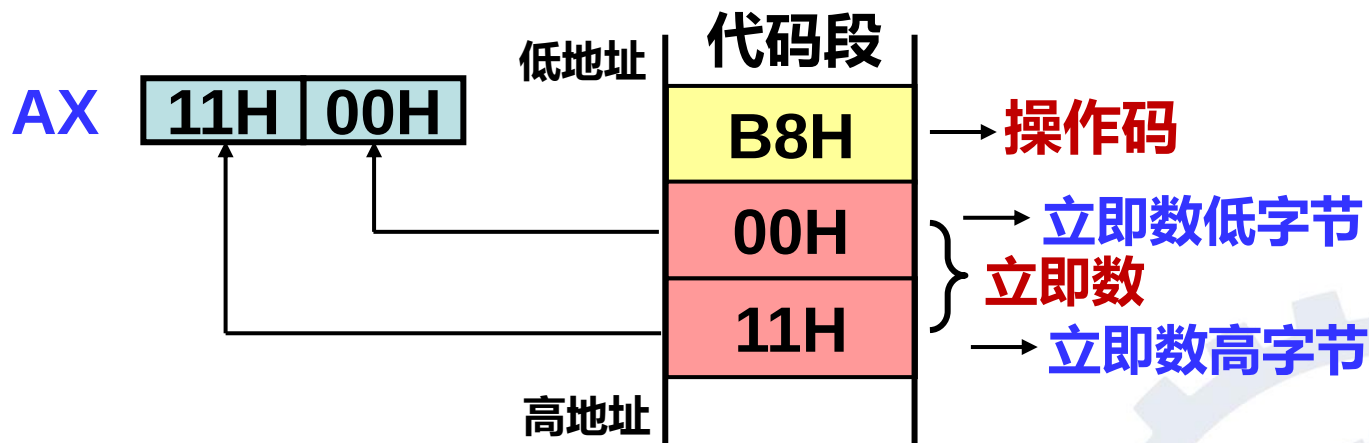
——存储器寻址





1. 立即数寻址 —— 操作数在指令中

MOV AX, 1100H



- 操作数从指令代码中立即得到，即立即数 (Immediate)，用常量形式直接表达
- 因此操作数存储在 **代码段 (CS)** 中



例: **MOV** AX, 1234H ; 1234H是十六进制数

MOV AL, 5 ; 5是十进制数

MOV AL, 'A' ; 'A'是字符常数, 等于41H

注意!

- 只能用于**源操作数**

MOV 40H, AL ×

- * 源操作数和目的操作数的**字长必须一致**

MOV AH, 3064H ×





2. 寄存器寻址 —— 操作数在寄存器中

MOV AX, BX

- 操作数在**寄存器**中，不需要访问存储器，速度快

注意！

*源操作数和目的操作数的**字长必须一致**

MOV AH, BX ×

*CS不能用MOV指令改变

MOV CS, AX ×





3. 存储器寻址

- 操作数在**存储器**中，用**存储单元的地址**表示
- 编程时使用包含**段寄存器和偏移地址**的**逻辑地址**
 - **段寄存器指示段基地址**
 - **偏移地址**由各种寻址方式计算，常被称为**有效地址EA**
(Effective Address)
- 一般情况下**存储器地址**由**有效地址**表示，**段寄存器不用显式说明**，**数据在默认的段中**。如果不使用默认段寄存器，则要用**段超越指令前缀**加以说明



(1) 段寄存器的默认和超越

访问存储器的方式	默认	可超越	偏移地址
取指令	CS	无	IP
堆栈操作	SS	无	SP
一般数据访问	DS	CS ES SS FS GS	有效地址EA
BP基址的寻址方式	SS	CS ES DS FS GS	有效地址EA
串操作的源操作数	DS	CS ES SS FS GS	SI
串操作的目的操作数	ES	无	DI



(2) 偏移地址的组成

- 16位有效地址

EA = 基址寄存器 + 变址寄存器 + 位移量

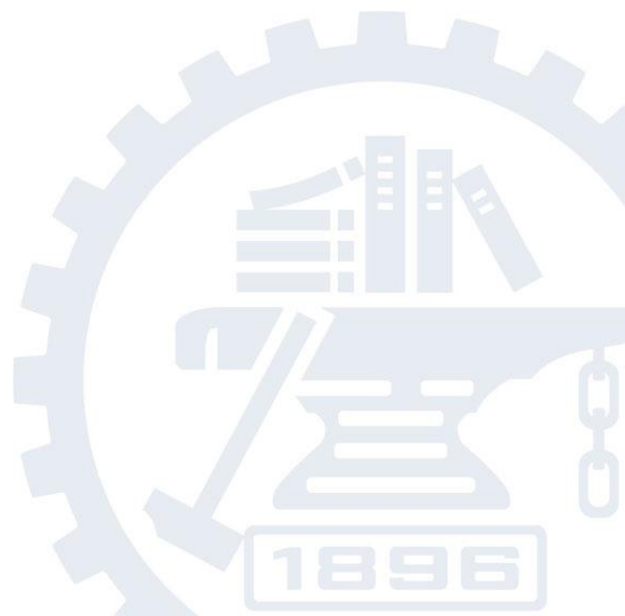
- 基址寄存器：BX或BP
- 变址寄存器：SI或DI
- 位移量：8或16位有符号值

- 一般情况下，由基址寄存器决定哪个段寄存器作为段指针（如果不使用段超越前缀）



(3) 存储器寻址的5种方式

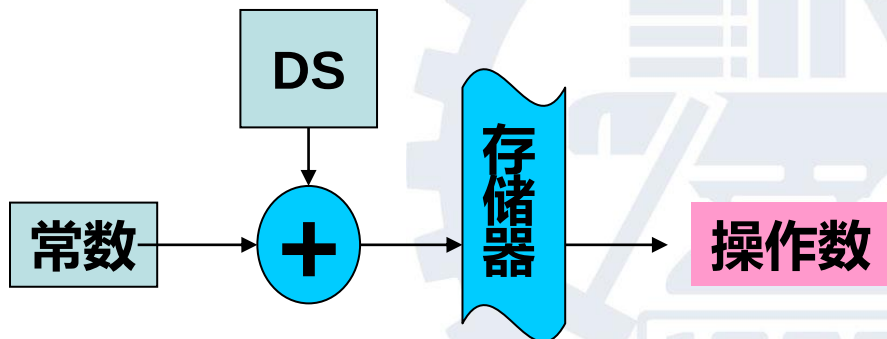
- ① 直接寻址
- ② 寄存器间接寻址
- ③ 寄存器相对寻址
- ④ 基址变址寻址
- ⑤ 相对基址变址寻址





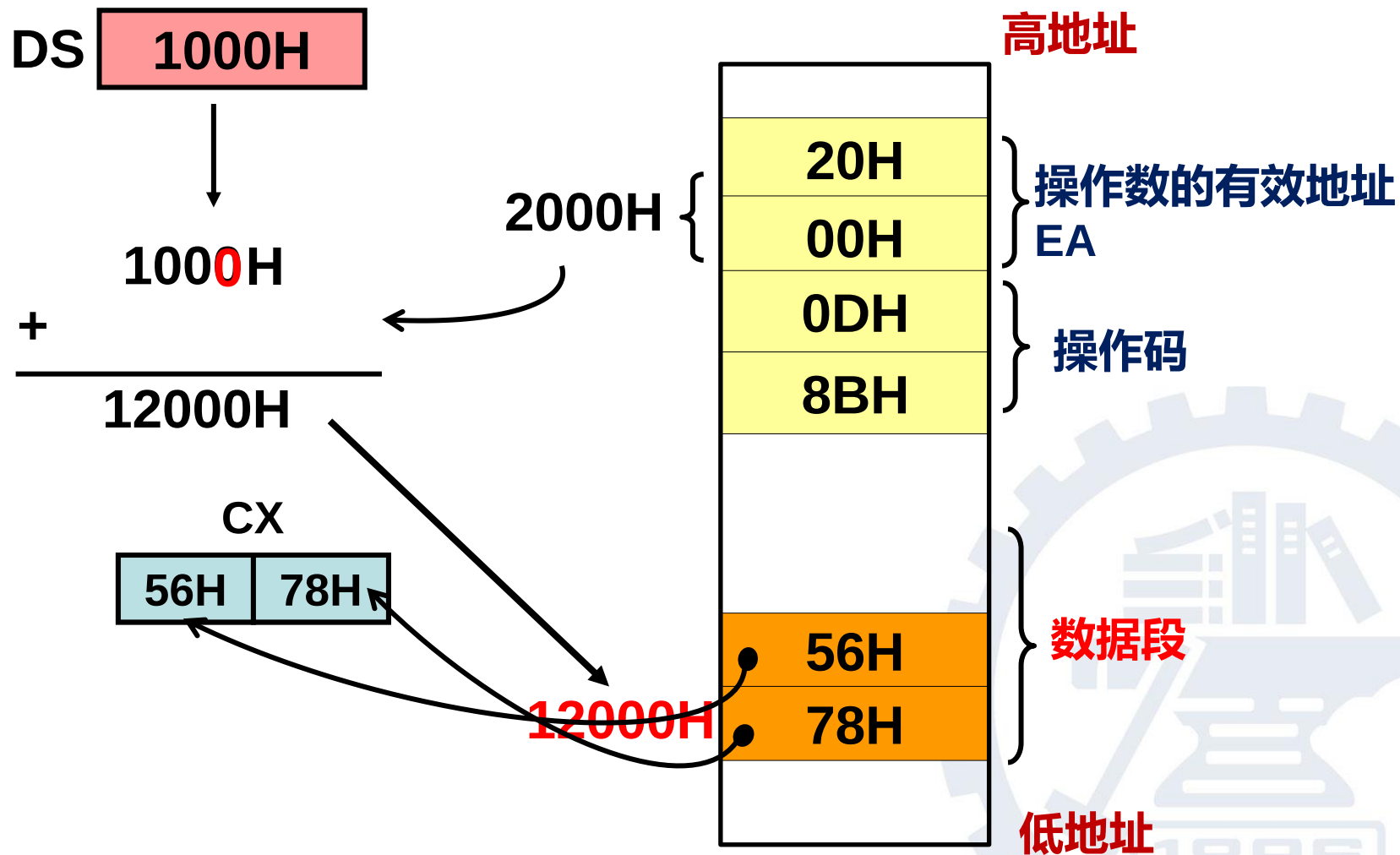
① 直接寻址 例：MOV CX, [2000H]

- 指令中直接包含了操作数的有效地址EA，在指令操作码之后
- 默认段地址在DS寄存器中，即操作数的实际地址（物理地址）是DS:EA
- 常用于存取变量





例：MOV CX, [2000H]





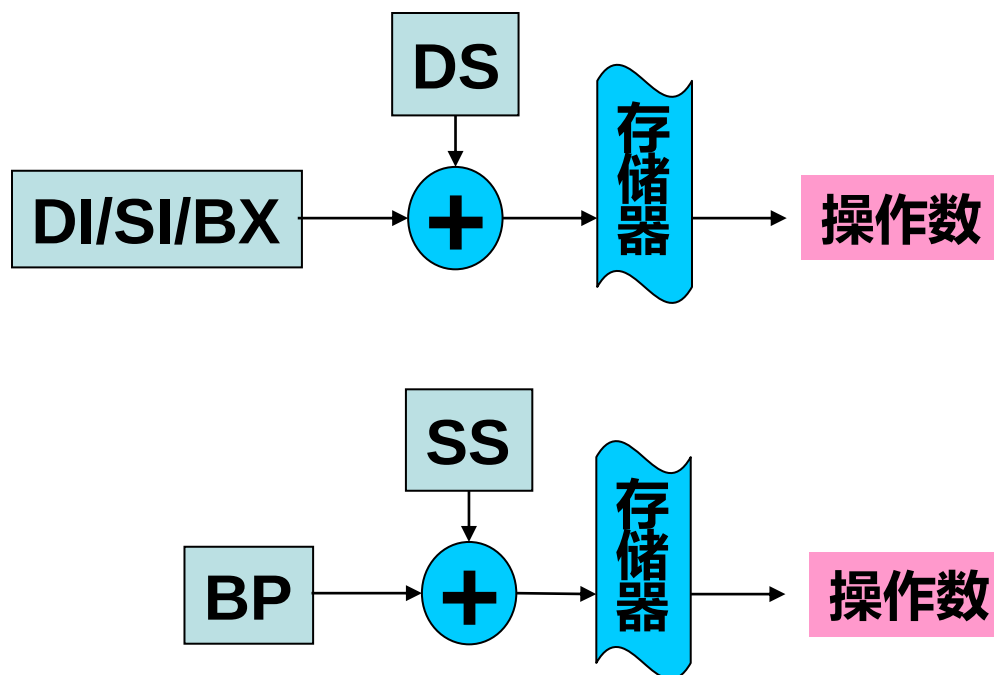
② 寄存器间接寻址

- 操作数的有效地址EA存放在**基址寄存器或变址寄存器**。不能放在AX\CX\DX中
- 可以方便地对数组的元素或字符串的字符进行操作
- 寄存器间接寻址没有说明存储单元类型



② 寄存器间接寻址 例: **MOV AX, [SI]**

- 操作数的段地址（数据处于哪个段）取决于选择哪一个间址寄存器：



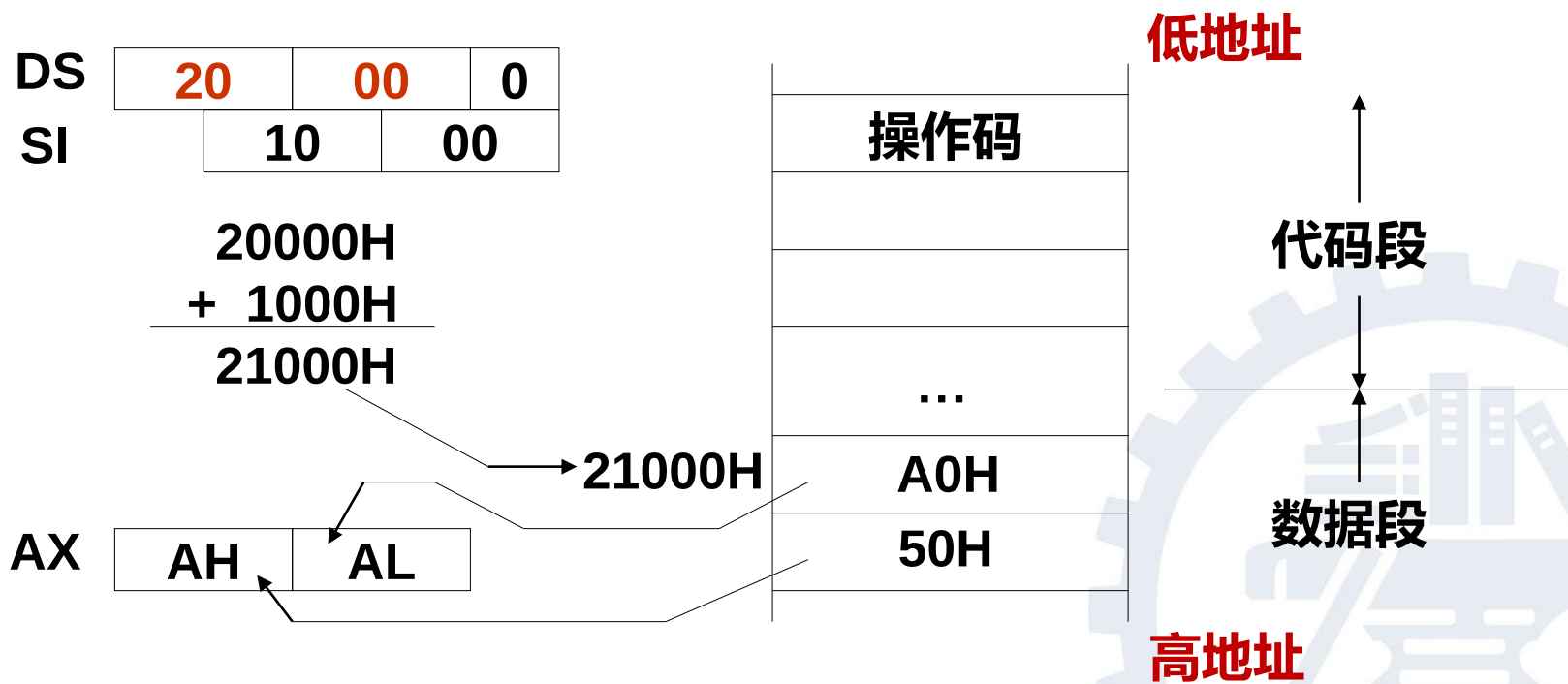
默认在数据段

默认在堆栈段



例1: **MOV** AX, [SI]

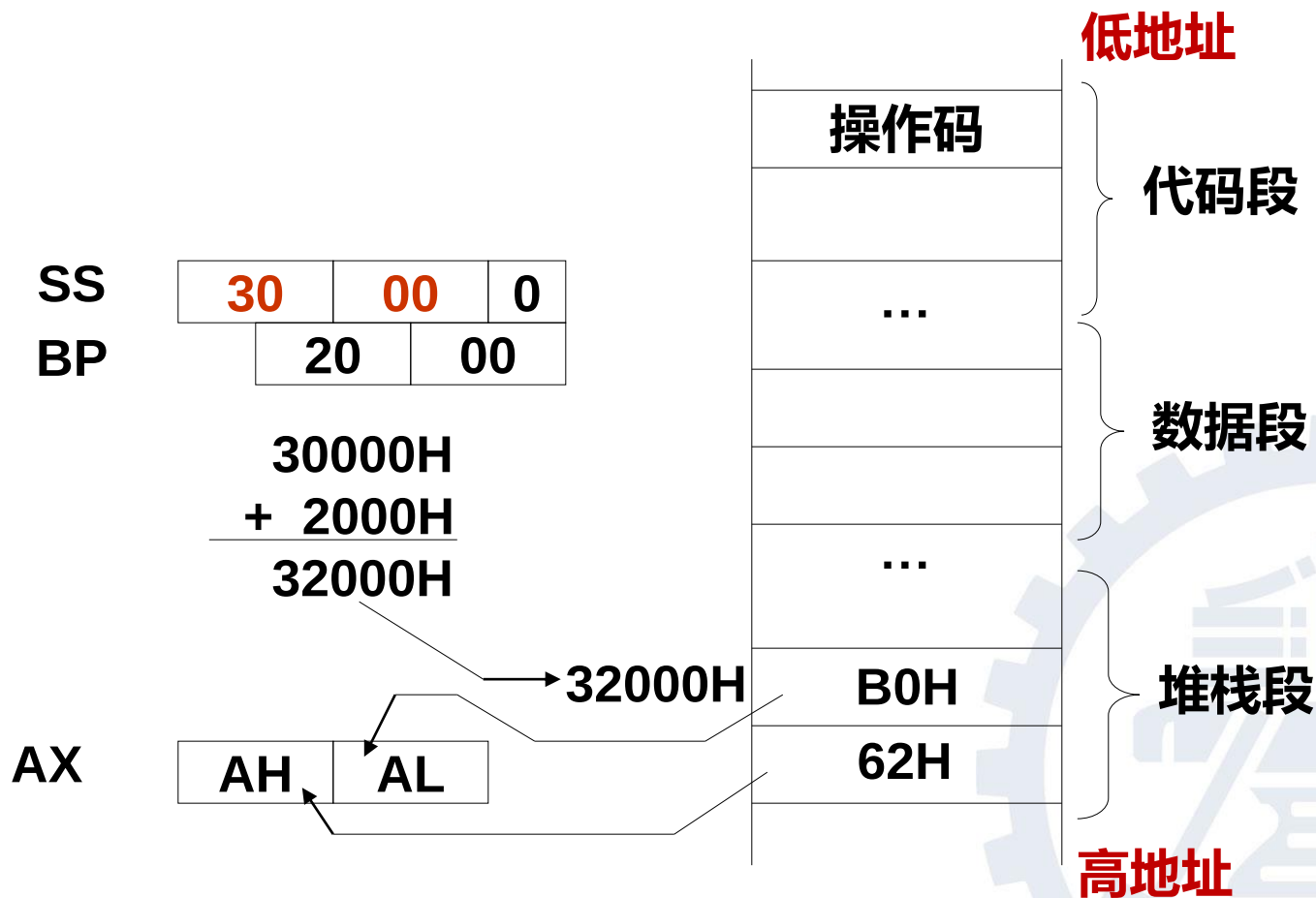
段地址在数据段DS





例2: MOV AX, [BP]

段地址在堆栈段SS





③ 寄存器相对寻址 例: **MOV** BX, NUM+[DI]

- 有效地址是寄存器内容与位移量之和
- 适用于数组、字符串、表格的操作

有效地址

$$\text{有效地址} = \begin{cases} (\text{BX}) \\ (\text{BP}) \\ (\text{SI}) \\ (\text{DI}) \end{cases} + \begin{cases} 8\text{位} \\ 16\text{位} \end{cases} \text{位移量}$$

DI/SI/BX+常数

BP+常数

DS

+

存储器

操作数

SS

+

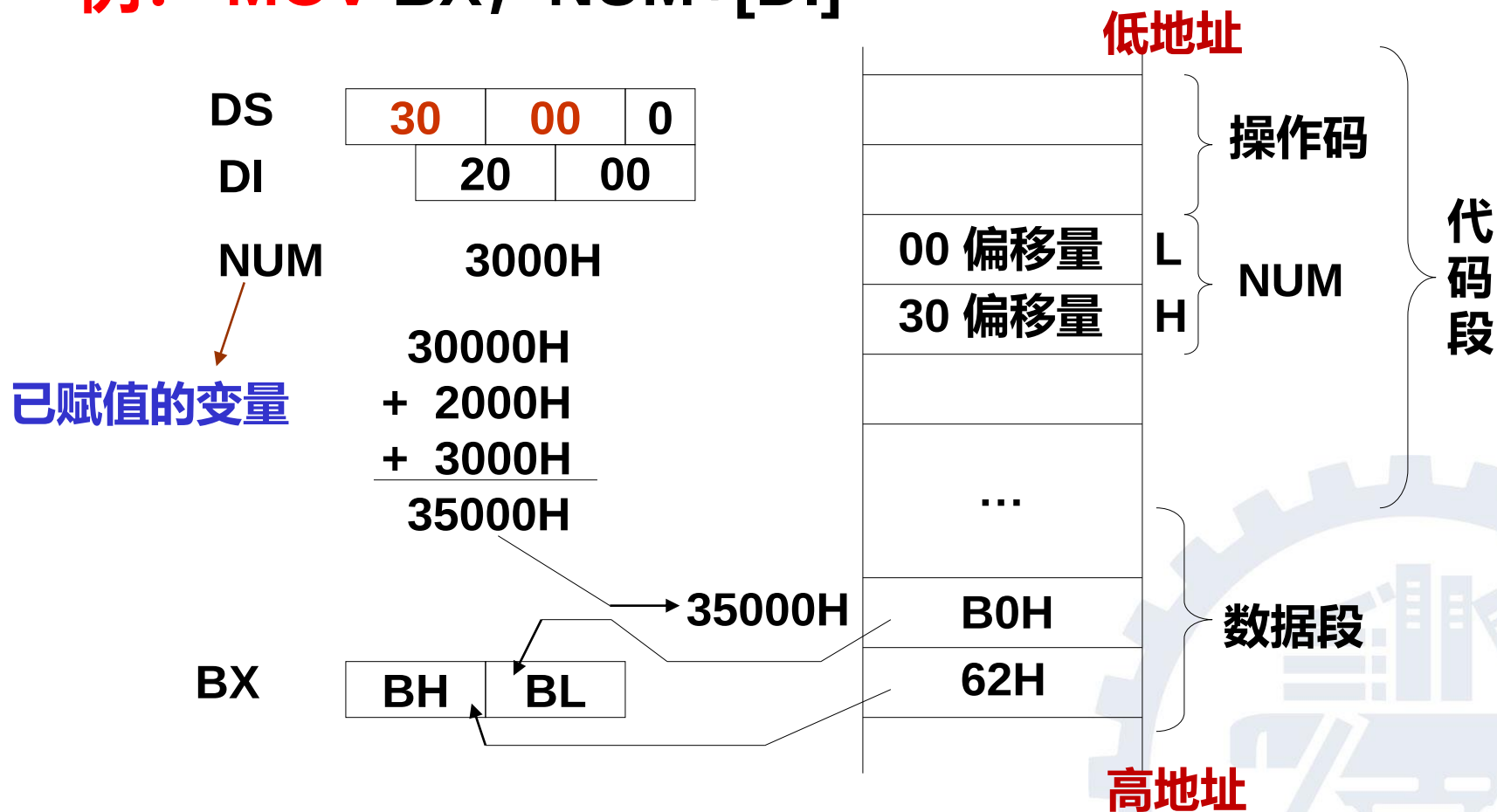
存储器

操作数



例: **MOV BX, NUM+[DI]**

段地址在数据段DS



上指令可以写成 **MOV BX, NUM[DI]**



④ 基址变址寻址

- 有效地址是基址寄存器和变址寄存器之和

$$\text{有效地址} = \begin{cases} (\text{BX}) \\ (\text{BP}) \end{cases} + \begin{cases} (\text{SI}) & \text{段地址在数据段DS} \\ (\text{DI}) & \text{段地址在堆栈段SS} \end{cases}$$

- 适用于二维数组、字符串、表格的操作
- 必须是一个基址寄存器和一个变址寄存器的组合

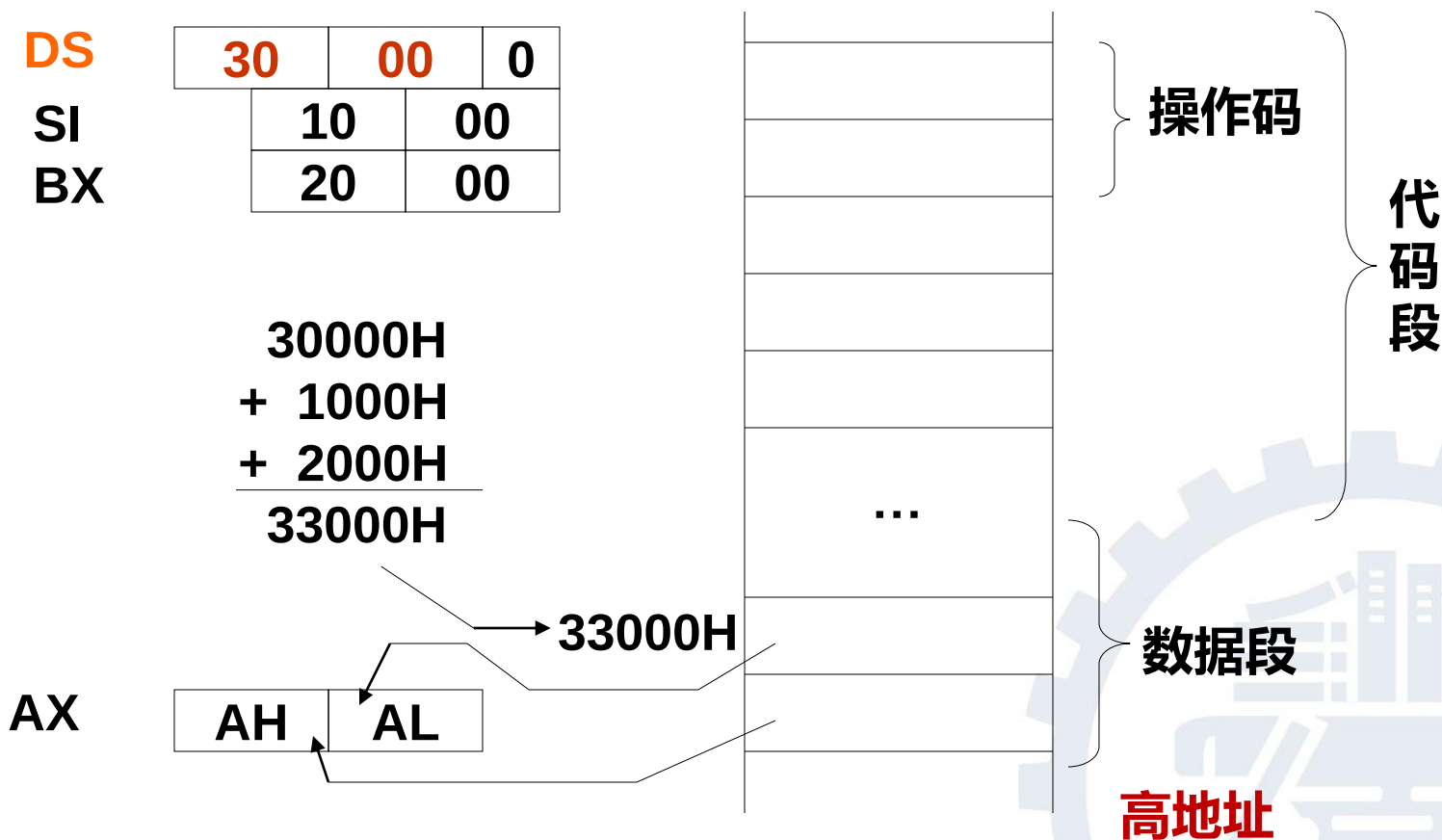
MOV AX, [BX][BP] ×

MOV AX, [SI][DI] ×



例1: MOV AX, [BX][SI]

段地址在数据段DS
低地址

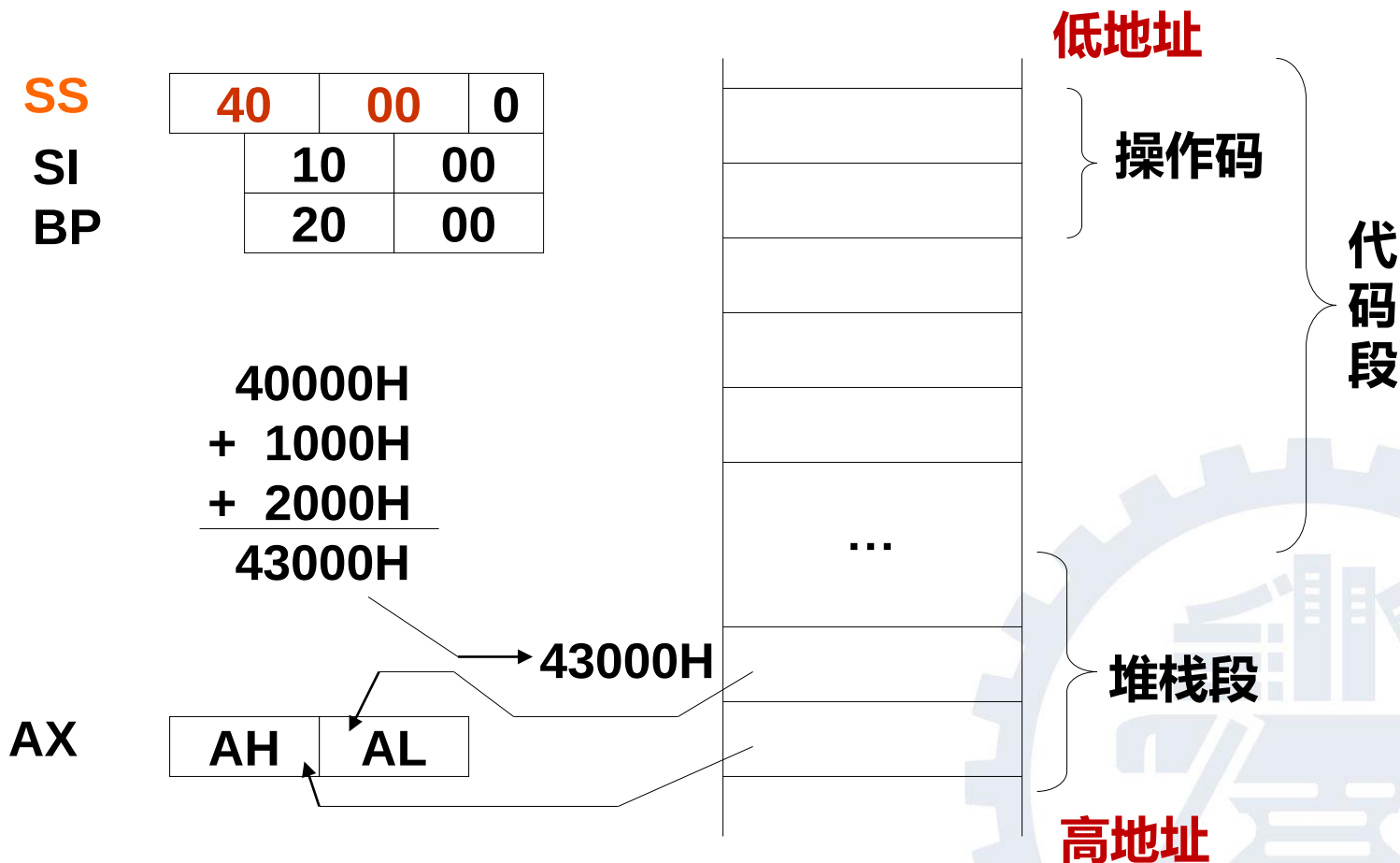


上指令可以写成 **MOV AX, [BX+SI]**



例2: **MOV AX, [BP][SI]**

段地址在堆栈段SS



上指令可以写成 **MOV AX, [BP+SI]**



⑤ 相对基址变址寻址方式

• 有效地址 =

$$\begin{cases} (\text{BX}) \\ (\text{BP}) \end{cases} + \begin{cases} (\text{SI}) \\ (\text{DI}) \end{cases} + \begin{cases} 8\text{位} \\ 16\text{位} \end{cases} \quad \text{位移量}$$

段地址在数据段DS

段地址在堆栈段SS



例: **MOV AX, MASK[BX][SI]**

段地址在数据段DS

低地址

DS	30	00	0
SI	10	00	
BX	20	00	
MASK	0250H		

30000H
+ 1000H
+ 2000H
+ 0250H

33250H

AX

AH AL

33250H

50H 偏移量
02H 偏移量

L
H

操作码

MASK

代码段

数据段

高地址

上指令可以写成 **MOV AX, [BX+SI+MASK]**
或 **MOV AX, MASK[BX+SI]**



操作数的寻址方式总结

立即数寻址

MOV AX , 3069H

寄存器寻址

MOV AL , BH

直接寻址

MOV AX , [2000H]

寄存器间接寻址

MOV AX , [BX]

寄存器相对寻址

MOV AX , NUM [SI]

基址变址寻址

MOV AX , [BP][DI]

相对基址变址寻址

MOV AX , MASK [BX][SI]



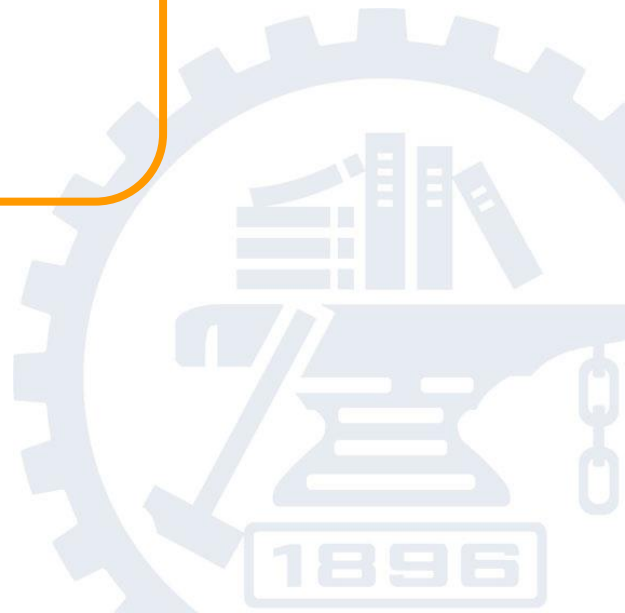
北京交通大学
BEIJING JIAOTONG UNIVERSITY



主要内容

3.1.1 操作数寻址

3.1.2 指令寻址





1. 段内直接寻址

- 转向的有效地址 = 当前(IP) + 位移量(8bit/16bit)

转向的有效地址与当前IP值之差

例: `JMP NEAR PTR NEXT` 近转移 -32768 ~ +32767

汇编时, 位移量为16位

`JMP SHORT NEXT` 短转移 -128 ~ +127

汇编时, 位移量为8位



2. 段内间接寻址 **CS不变，但IP变化。**

- 转向的有效地址是一个寄存器或存储单元的内容
- 可用除立即数以外的任何一种数据寻址方式得到

例： TABLE = 20A1H (BX) = 1256H (SI) = 528FH (DS) = 2000H
(232F7H) = 3280H (264E5H) = 2450H

JMP BX ; (IP)=1256H

JMP TABLE[BX]

JMP WORD PTR TABLE[BX] ; (IP)=(232F7H) = 3280H

JMP [BX][SI]

JMP WORD PTR [BX][SI] ; (IP)=(264E5H) = 2450H



例：JMP WORD PTR TABLE[BX]

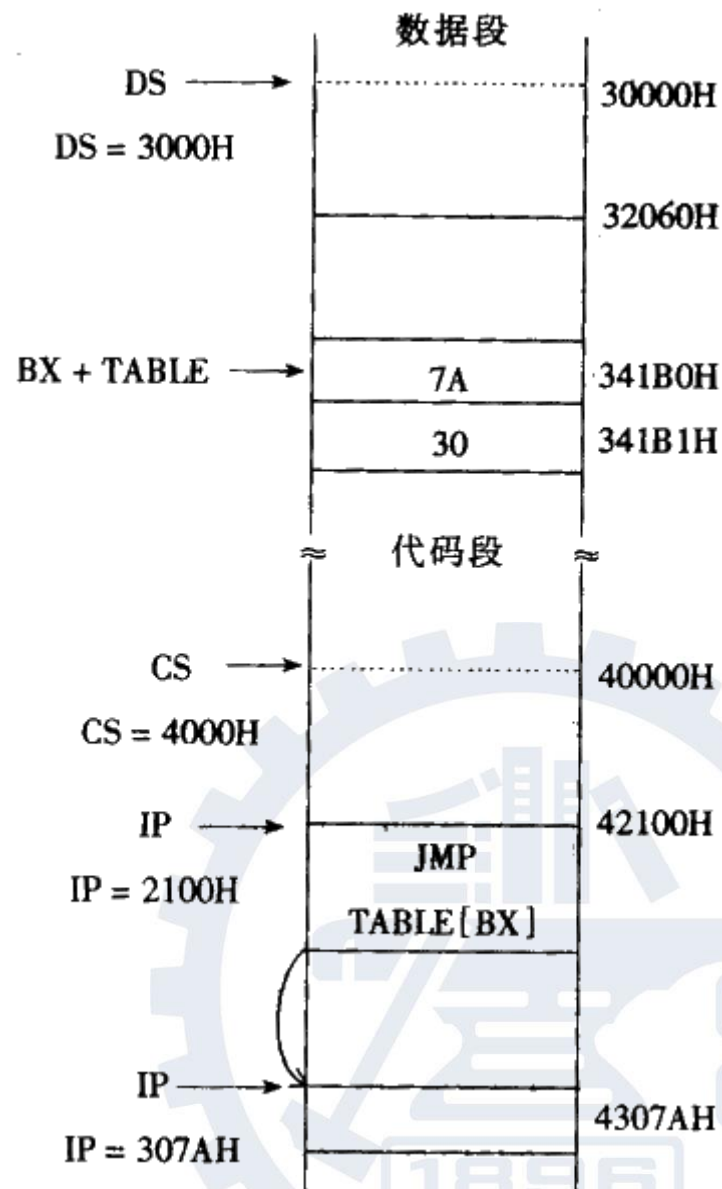
或 JMP TABLE[BX]

如 $DS = 3000H$, $BX = 2150H$,
 $TABLE = 2060H$, 则操作数存储单元
的物理地址 $= 16 \times DS + TABLE + BX$

$= 30000 + 2060 + 2150 = 341B0H$

执行前, $IP = 2100H$, $(341B0H)$
 $= 7AH$, $(341B1H) = 30H$ 。

执行后, $IP = 307AH$, $(341B0H)$
 $= 7AH$, $(341B1H) = 30H$ 。





3. 段间寻址 CS与IP都变化(一般仅供OS使用)

- 转向的有效地址是XXXX:YYYY形式

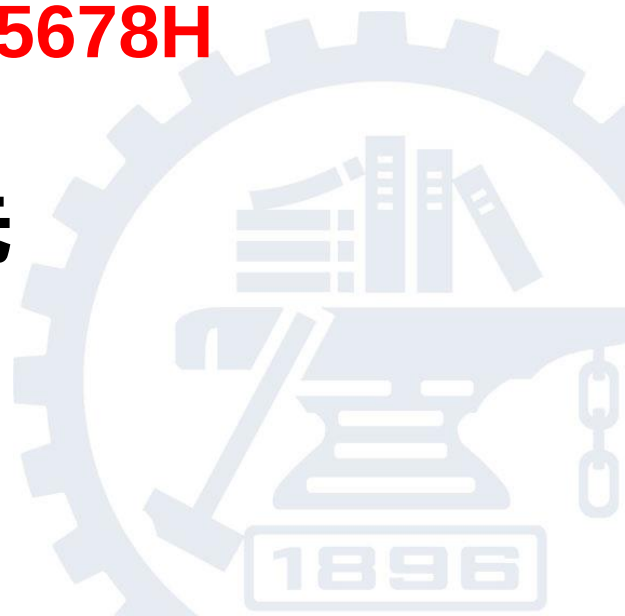
例： JMP 1234H:5678H

； (CS)=1234H , (IP)=5678H

JMP dword ptr 内存地址单元

； (CS)=(内存地址单元+2) ,

(IP)=(内存地址单元)





1、现有 $(DS) = 2000H$, $(BX) = 0100H$, $(SI) = 0002H$,
 $(20100) = 12H$, $(20101) = 34H$, $(20102) = 56H$,
 $(20103) = 78H$, $(21200) = 2AH$, $(21201) = 4CH$,
 $(21202) = 0B7H$, $(21203) = 65H$, 试说明下面各条指令
执行后AX寄存器的内容。

- (1) `MOV AX, 1200H`
- (2) `MOV AX, BX`
- (3) `MOV AX, [1200H]`
- (4) `MOV AX, [BX]`
- (5) `MOV AX, 1100H[BX]`
- (6) `MOV AX, [BX][SI]`
- (7) `MOV AX, 1100H[BX][SI]`



作业



2、假设 $(DS) = 2000H$, $(ES) = 2100H$, $(SS) = 1500H$, $(SI) = 00A0H$, $(BX) = 0100H$, $(BP) = 0010H$, 数据段中的变量名 VAL 的偏移地址值为 $0050H$, 试指出下列源操作数的寻址方式是什么? 其物理地址值是什么?

- (1) MOV AX, 0BAH
- (2) MOV AX, [100H]
- (3) MOV AX, [BX]
- (4) MOV AX, [BP]
- (5) MOV AX, [BX+10]
- (6) MOV AX, [BX][SI]
- (7) MOV AX, BX
- (8) MOV AX, VAL
- (9) MOV AX, ES: [BX]
- (10) MOV AX, [SI]
- (11) MOV AX, VAL [BX]
- (12) MOV AX, VAL [BX][SI]



作业