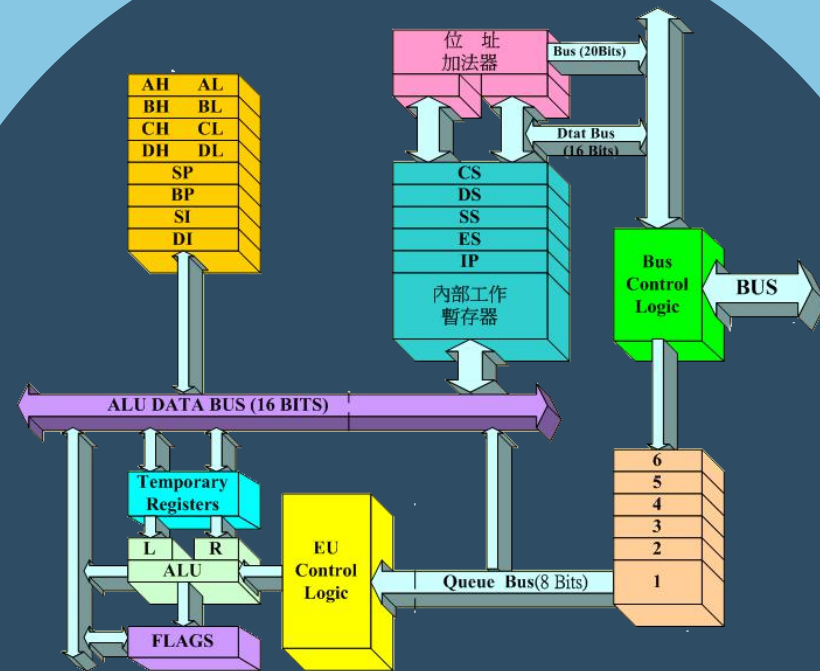


[bx+si]和[bx+di]方式寻址

贺利坚 主讲



汇编语言程序设计
Assembly Language

$[bx+si]$ 和 $[bx+di]$ 方式指定地址

🖥️ $[bx+si]$ 表示一个内存单元

📁 偏移地址为 $(bx)+(si)$ （即 bx 中的数值加上 si 中的数值）。

🖥️ 指令 $\text{mov ax}, [bx+si]$ 的含义

📁 将一个内存单元的内容送入 ax

📁 这个内存单元的长度为2字节（字单元），存放一个字

📁 偏移地址为 bx 中的数值加上 si 中的数值

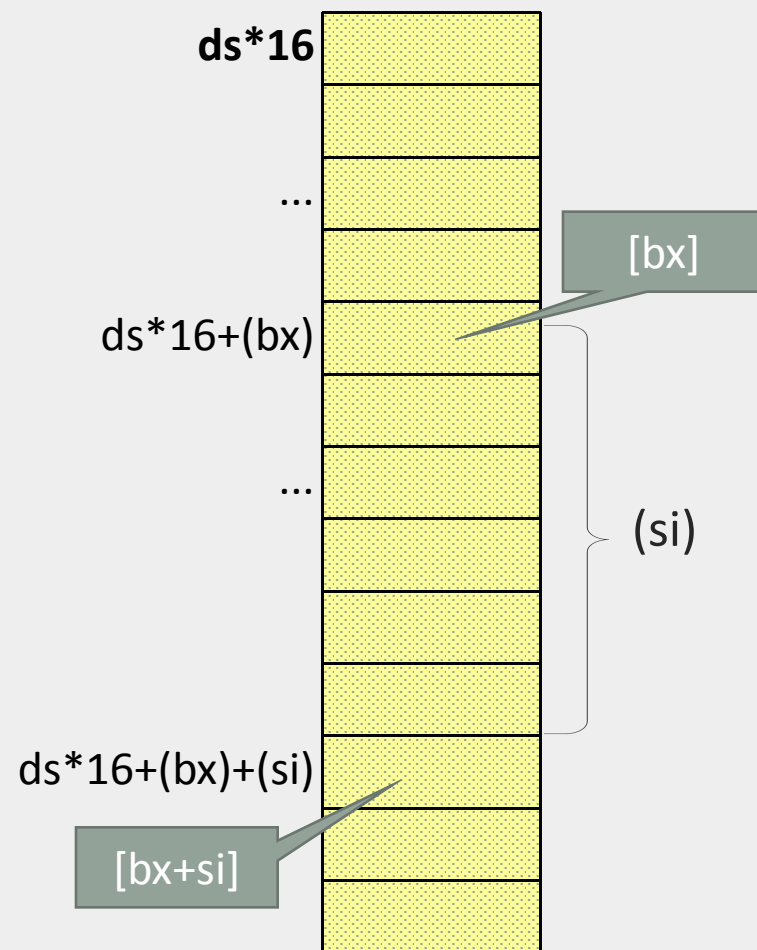
📁 段地址在 ds 中

🖥️ 指令 $\text{mov ax}, [bx+si]$ 的数学化的描述

📁 $(ax) = ((ds) * 16 + (bx) + (si))$

🖥️ $\text{mov ax}, [bx+si]$ 的其他写法

📁 $\text{mov ax}, [bx][si]$



应用案例

💻 内存中数据 2000:1000 BE 00 06 00 00 00

💻 程序执行后，ax、bx、cx中的内容？

DS		SI		DI	
AX		BX		CX	

```
mov ax,2000H
mov ds,ax
mov bx,1000H
mov si,0
mov ax,[bx+si]
inc si
mov cx,[bx+si]
inc si
mov di,si
mov ax,[bx+di]
```

```
C:\>debug
-a
073F:0100 mov ax, 2000
073F:0103 mov ds, ax
073F:0105 mov bx, 1000
073F:0108 mov si, 0
073F:010B mov ax, [bx+si]
073F:010D inc si
073F:010E mov cx, [bx+si]
073F:0110 inc si
073F:0111 mov di, si
073F:0113 mov ax, [bx+di]
073F:0115
-
-e 2000:1000 BE 00 06 00 00 00
-g 0113
AX=00BE BX=1000 CX=0600 DX=0000 SP=00FD BP=0000 SI=0002 DI=0002
DS=2000 ES=073F SS=073F CS=073F IP=0113  NU UP EI PL NZ NA PO NC
073F:0113 8B01          MOV     AX,[BX+DI]          DS:1002=0006
-t
AX=0006 BX=1000 CX=0600 DX=0000 SP=00FD BP=0000 SI=0002 DI=0002
DS=2000 ES=073F SS=073F CS=073F IP=0115  NU UP EI PL NZ NA PO NC
073F:0115 0000          ADD     [BX+SI],AL          DS:1002=06
```