寄存器寻址

寻址方式（Addressing）

- 通过地址访问数据或指令
- 数据寻址：

  指令执行过程中，

  访问所需要操作的数据(操作数)

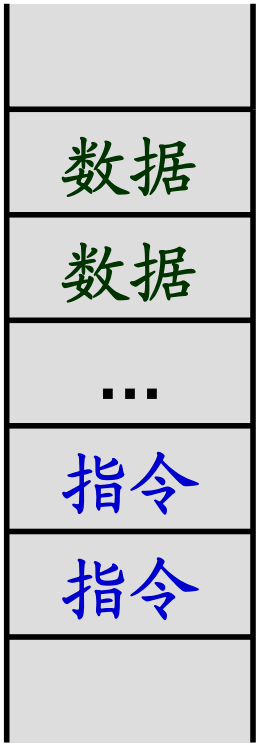
操作数在哪儿呢？

- 指令寻址：

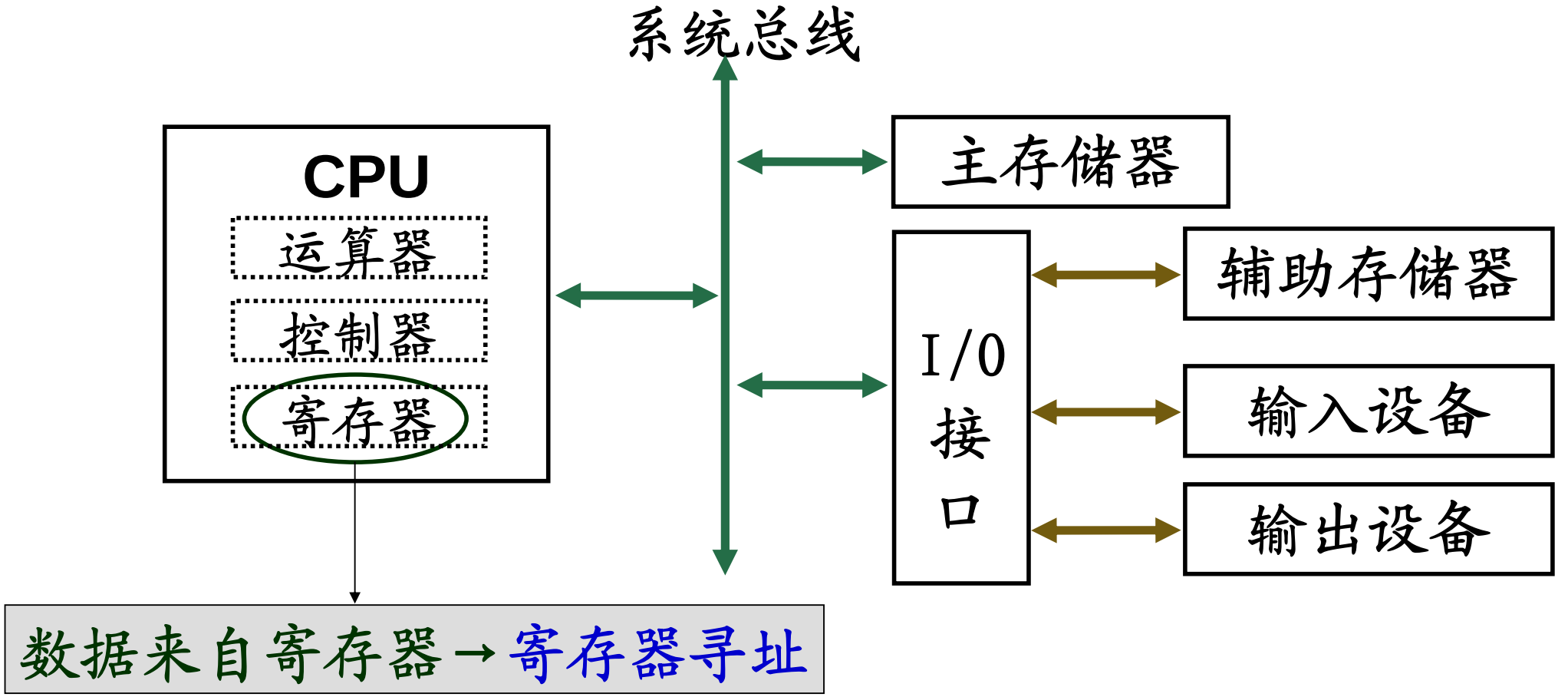
  一条指令执行后，

  确定执行的下一条指令的位置

指令又在哪儿呢？

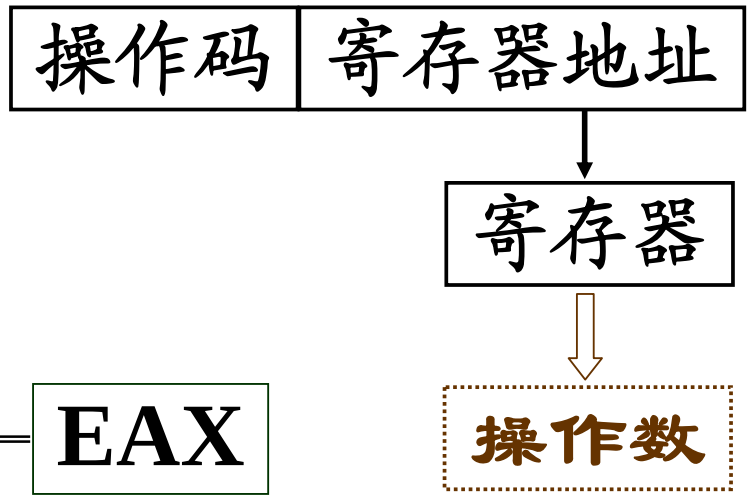


操作数在那儿！



寄存器寻址

- 操作数存放在处理器的内部寄存器中
  - 用寄存器名表示它的内容




```
MOV EBX, EAX
;目的操作数和源操作数
;均采用寄存器寻址
```

EBX ⟸ EAX

32位通用寄存器：EAX EBX ECX EDX ESI EDI EBP ESP

16位通用寄存器：AX BX CX DX SI DI BP SP

8位通用寄存器：AH AL BH BL CH CL DH DL

寄存器寻址程序

```
;代码段
mov al, ah
mov bx, ax
mov ebx, eax
mov dx, ds
mov es, dx
```

源操作数和目的操作数均是寄存器寻址

```
;代码段
mov al, 12
mov bx, 12
mov bvar, cl
mov wvar, dx
mov dvar, edx
```

目的操作数是寄存器寻址

源操作数是寄存器寻址

8位寄存器的类型是字节型
16位寄存器的类型是字型
32位寄存器的类型是双字型

语法错

```
mov edi,si
eg0209.asm(11) : error A2022:
        instruction operands must be the same size
```

错误语句的行号

错误编号

错误信息

- 常见语法错误原因
  - 拼写错误、多余的空格、遗忘的后缀字母或前导0、不正确的标点、太过复杂的常量或表达式
  - 操作数类型不匹配、错用寄存器……

本讲总结

- 寄存器寻址
  - 操作数存放在处理器的内部寄存器中
  - 用寄存器名表示它的内容
- 寄存器寻址方式简单快捷，最常使用
  - 绝大多数指令采用通用寄存器
  - 部分指令支持专用寄存器（例如段寄存器）

符号	含义
r8	8位寄存器
r16	16位寄存器
r32	32位寄存器
reg	通用寄存器