

# 一、转移指令

# 转移指令

通过修改指令的偏移地址或段地址及偏移地址实现程序的转移

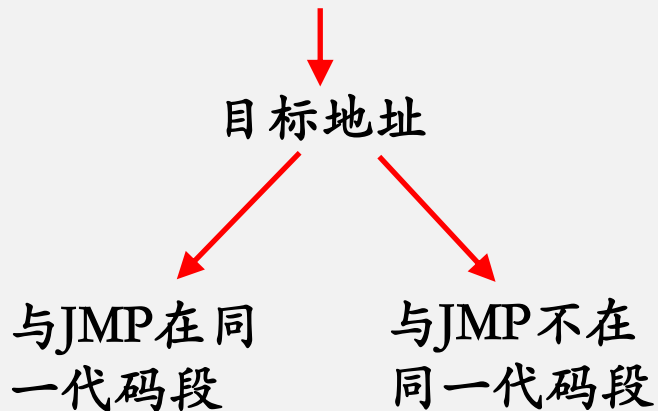
- 无条件转移指令 → 无条件转移到目标地址
- 条件转移指令 → 当具备一定条件时转移到目标地址

通常指状态标志位

# 1. 无条件转移指令

## ■ 格式:

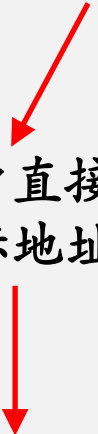
■ **JMP OPRD**



可以实现在当前  
代码段内或段间  
转移

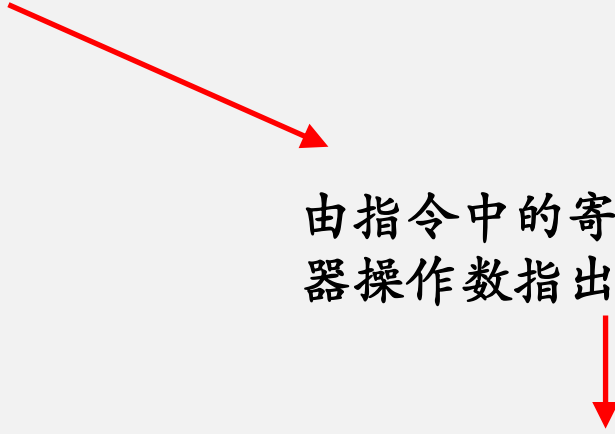
# 1) 无条件段内转移

- 转移的目标地址在当前代码段内，段地址不改变。
- 即：目标地址是16位偏移地址。



指令中直接给出目标地址

段内直接转移



由指令中的寄存器或存储器操作数指出目标地址

段内间接转移

# 段内直接转移

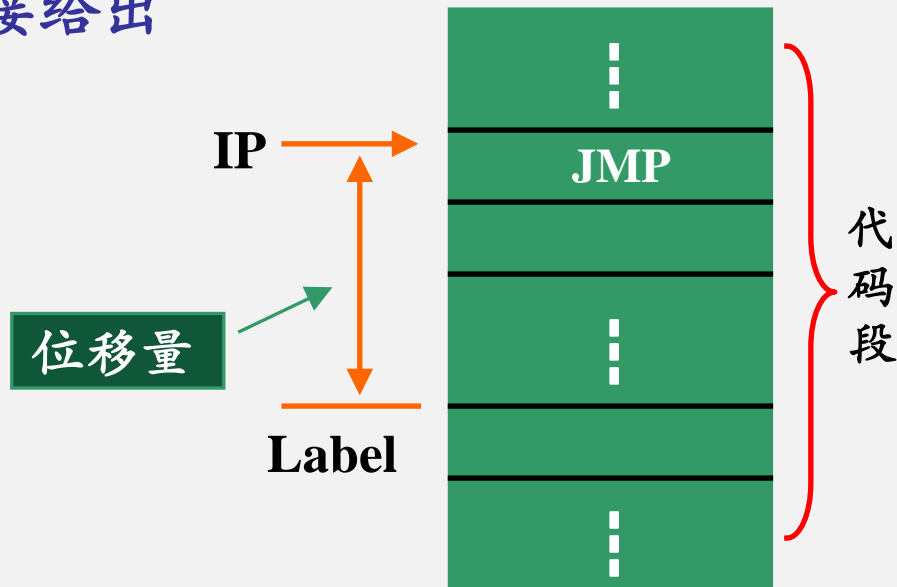
- 转移的目标地址由指令直接给出

- 格式:

- `JMP Label`



近地址标号



下一条要执行指令的偏移地址=当前IP+位移量

# 段内间接转移

- 段内间接转移

- 转移的目标地址存放在某个16位寄存器或存储器的某两个单元中

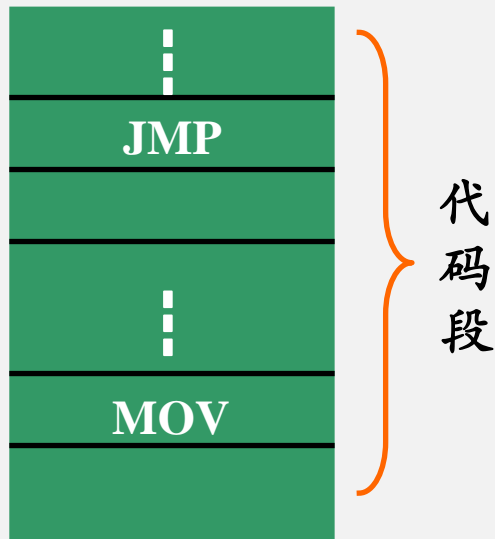
- 例：

- `MOV BX, 1200H`
- `JMP BX`

- 执行完上述指令后：

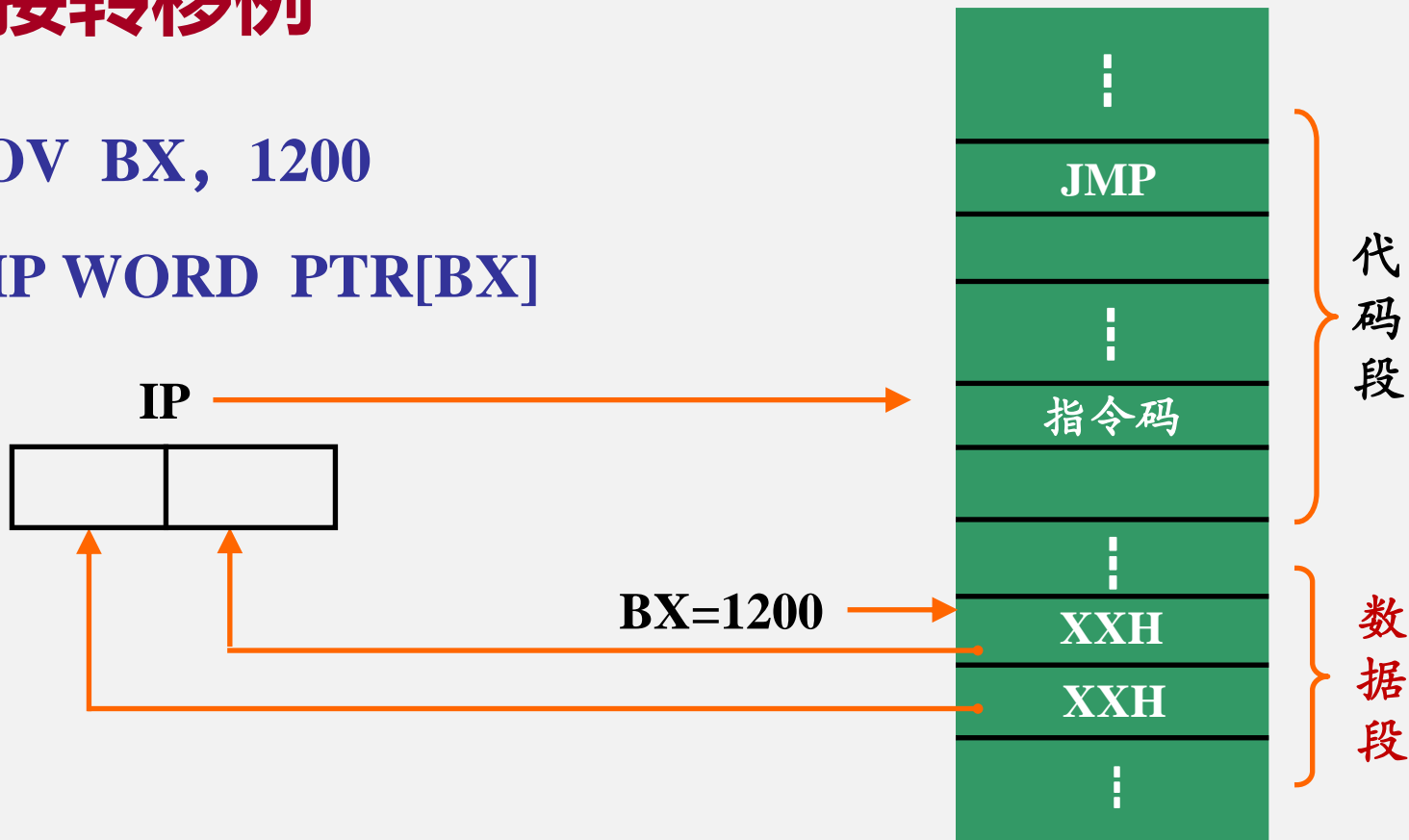
- `IP=1200H`

1200H



# 段内间接转移例

- `MOV BX, 1200`
- `JMP WORD PTR[BX]`



## 2) 无条件段间转移

- 转移的目标地址不在当前代码段内。
- 目标地址为32位，包括段地址和偏移地址。

指令中直接给出目标地址

段间直接转移

由指令中的32位存储器操作数指出目标地址

段间间接转移

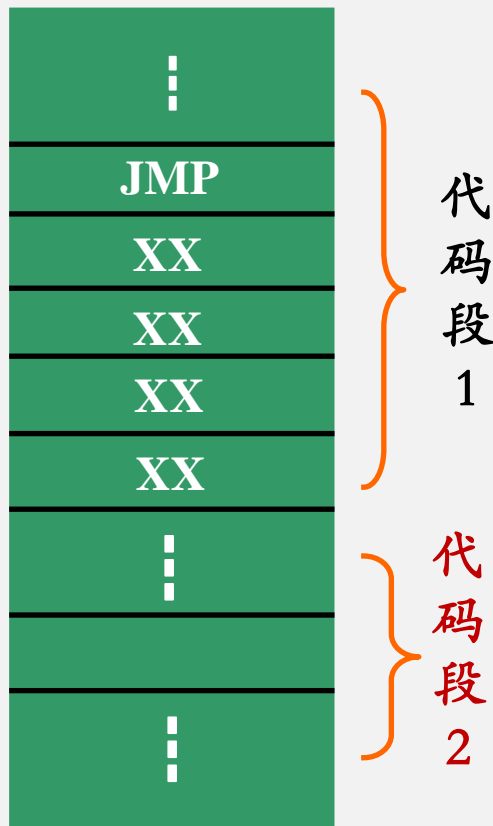


# 段间直接转移

- 段间直接转移
  - 转移的目标地址由指令直接给出
- 格式:
  - **JMP FAR Label**

远地址标号

Label



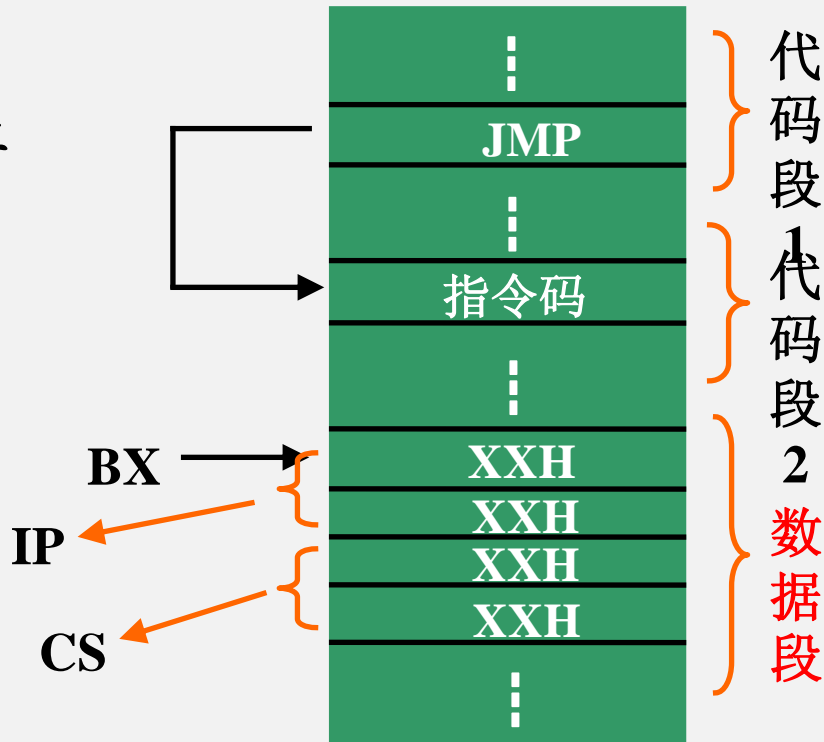
# 段间间接转移

## ■ 段间间接寻址

- 转移的目标地址由指令中的32位操作数给出
- 32位目标地址须存放于内存中

## ■ 例：

- **JMP DWORD PTR[BX]**



# 无条件转移指令例

**CS: IP**

**(1) 2000: 0100                    MOV AX,1200H**

**(2) 2000: 0103                    JMP NEXT**

**---**

**(3) 2000: 0120    NEXT: MOV BX,1200H**

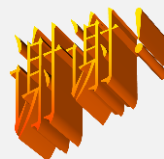
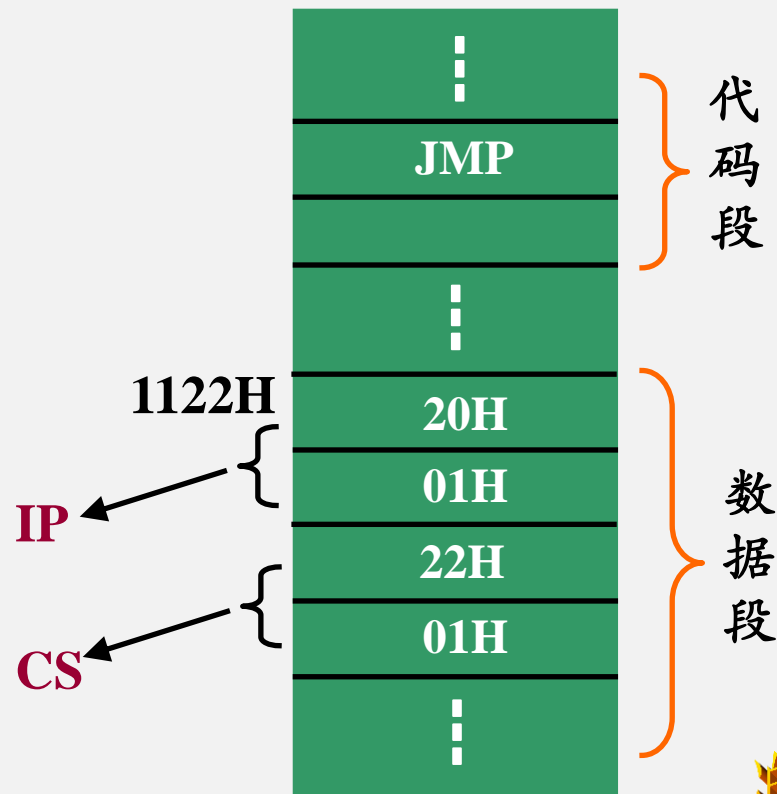
**(4)                                JMP BX**

**---**

**(5) 2000: 1200**

# 无条件转移指令例

- `MOV SI, 1122H`
  - `MOV WORD PTR[SI], 0120H`
  - `ADD SI, 2`
  - `MOV WORD PTR[SI], 0122H`
- `JMP DWORD PTR[SI-2]`**



# 程序代码

```
START: XOR AL, AL
        MOV PLUS, AL
        MOV MINUS, AL
        MOV ZERO, AL
        LEA SI, TABLE
        MOV CL, 100
        CLD
```

```
CHECK: LODSB
```

```
        OR AL, AL
```

```
        JS X1
```

```
        JZ X2
```

```
        INC PLUS
```

```
        JMP NEXT
```

```
X1: INC MINUS
```

```
        JMP NEXT
```

```
X2: INC ZERO
```

```
NEXT: DEC CL
```

```
        JNZ CHECK
```

```
        HLT
```