目 录

1 操作符 offset

offset 用于取得标号相对于段地址的偏移地址，使用格式为：

```
offset 标号
```

使用例子如下所示：

```
assume cs:codesg
codesg segment

start:
    mov ax, offset start
s:
    mov ax, offset s

codesg ends
end
```

2 jmp 指令

jmp 指令为无条件转移指令，可以只修改 IP，也可以同时修改 CS 和 IP。

jmp 指令要给出两种信息：转移的目的地址、转移的距离。

2.1 根据位移进行转移的 jmp 指令

“jmp short 标号”指令对应的机器码中，并不包含转移的目的地址，而包含转移的位移，其中这个位移是编译器根据汇编指令中的“标号”计算出来的。

“jmp short 标号”的功能为：(IP)=(IP)+8 位位移，这里的 8 位位移 =“标号”处的地址-jmp 指令后的第一个字节的地址。

“jmp near ptr 标号”功能类似，它的功能是：(IP)=(IP)+16 位位移，其中 16 位位移 =“标号”处地址-jmp 指令后的第一个字节的地址。

2.2 转移的目的地址在指令中的 jmp 指令

“jmp far ptr 标号”指令中包含了标号的段地址喝偏移地址，用于修改 CS 和 IP，(CS)= 标号所在段的段地址，(IP)= 标号在段中的偏移地址。

2.3 转移的目的地址在寄存器的 jmp 指令

“jmp 16 位寄存器”，它的功能是：(IP)=(16 位寄存器)。

2.4 转移的目的地址在内存中的 jmp 指令

“jmp word ptr 内存单元地址”，它的功能是：(IP)=(内存单元地址)。

“jmp dword ptr 内存单元地址”，它的功能是：(CS)=(内存单元地址 +2)，(IP)=(内存单元地址)。也就是说，从内存单元地址处开始存放两个字，高地址处的字是转移的目的段地址，低地址处的字是转移的目的偏移地址。

3 jcxz 指令

jcxz 指令是有条件转移指令，所有的有条件转移指令都是段转移，对 IP 的修改范围为：-128 ～ 127。指令格式为：

```
jcxz 标号
```

指令功能为：当 (cx)=0 时，(IP)=(IP)+8 位位移，其中 8 位位移 =“标号”处的地址-jmp 指令后的第一个字节的地址。

4 loop 指令

loop 指令为循环指令，所有的循环指令都是短转移，指令格式如下：

```
loop 标号
```

loop 指令的功能为：1.(cx)=(cx)-1；2. 如果 (cx) 不等于零，就跳转到标号处执行指令。