3.10 循环控制指令

- 循环程序的组成

一个标准的循环程序应由以下四部分组成：

- 初始化准备部分

  为循环做准备，不在循环体内。

- 循环工作部分

  是循环程序解题所需的核心程序，题目要完成的功能在此进行。根据题目要求不同，这部分程序可以很简单，也可以很复杂，甚至是内循环嵌套外循环结构——多重循环程序。

3.10 循环控制指令

- 参数调整部分

  主要用来更新某些数据或修正循环控制的参数，以保证每次循环所完成的功能不是完全重复的。

- 出口判定部分

  循环程序中至少要有一个出口判定，以保证循环程序正常结束。也有些循环程序有多个出口，程序可以有多个条件作为循环结束的控制，只要其中一个条件满足即可结束循环。

注： 从程序设计上说，以上四个部分的分界可能不是很明确的，有时工作部分与调整部分可能就是同一段程序，但从功能上说，以上几个部分都是必需的。

3.10 循环控制指令

- 循环控制指令

为了便于循环控制，8086/8088CPU 专门设置了一类循环控制类指令:

格式:

LOOP LABEL ；(CX)←(CX)-1, (CX)≠0时转LABEL

LOOPZ/LOOPE LABEL

；(CX)←(CX)-1, (CX)≠0且ZF=1时转LABEL

LOOPNZ/LOOPNE LABEL

；(CX)←(CX)-1, (CX)≠0且ZF=0时转LABEL

JCXZ LABEL ；CX=0时转LABEL

注：循环控制指令的寻址方式均为段内直接转移，而且为短转移方式

3.10 循环控制指令

1. LOOP指令

是常用的循环控制指令，（CX）的内容为设定的循环次数，每循环一次（CX）的内容减1，直到（CX）为零时退出循环。其循环结构为：

```
        MOV  CX, 次数
                ；循环准备
标号：
                ；循环体
        LOOP  标号
```

这里的“标号”与“LOOP 标号”指令之间至多包含128字节，这是因为循环控制指令为短转移指令。

3.10 循环控制指令

2. LOOPZ/LOOPE指令

与LOOP指令类似，只是当(CX)≠0且ZF=1时才转至LABEL，因此是否循环，除了与设定的循环次数有关，还与循环中设定的条件是否满足有关。

程序结构：

```
        MOV  CX, 次数
                ；循环准备
   L1：
                ；循环体
        CMP  AX,BX ；若(AX)≠(BX)，即ZF=0,则退出循环
        LOOPZ  L1
```

适合于在指定区域中查找不同的“字符”，当找到不同的“字符”时，会自动退出循环。

3.10 循环控制指令

3. LOOPNZ/LOOPNE

功能与LOOPZ/LOOPE指令相反，只是当(CX)≠0且ZF=0时才转至LABEL。

程序结构：

```
        MOV  CX, 次数
                ；循环准备
   L2：
                ；循环体
        CMP  AX,BX ；若(AX)＝(BX) ，即ZF=1，则退出循环
        LOOPNZ  L2
```

适合于在区域中查找指定的“字符”，当找到指定的“字符”时，会自动退出循环。

3.10 循环控制指令

4. JCXZ指令

经常与循环指令配合使用。如果循环次数（CX）设定为0，则要循环65536次（最大的循环次数），这一点与常规表示不同。为此，应该在进入循环前检查（CX）的值，如果（CX）为0，则跳过循环，这时可以采用JCXZ指令完成，其结构为：

```
        MOV   CX, VAR1        ；设定循环次数，可能为0
        JCXZ  DONE            ；当（CX）＝0时，跳过循环
          ；循环准备
L3:
          ；循环体
        LOOP  L3
DONE:
```

3.10 循环控制指令

例. 在BUFFER中保存有15个无符号字节型数据，编写程序产生这组数据的校验和，并置入第16个字节单元中。

在数据段中已经定义好字节型变量BUFFER，常用的校验和产生算法: $h = \sum_i x_i \ MOD\ 256$

程序片段:

```
        LEA   SI, BUFFER
        MOV   CX, 15
        MOV   AL, 0
L1:
        ADD   AL, [SI]
        INC   SI
        LOOP  L1
        MOV   [SI], AL      ；保存校验和
```