****

**汇编语言程序设计课程设计**

****

|  |  |
| --- | --- |
| **题目** | **分支与循环结构程序设计** |
| **学院** | **计算机学院** |
| **专业** | **计算机科学与技术** |
| **学号** |  |
| **姓名** |  |
| **指导老师** | **余波老师** |
| **实验时间** | **2022年5月24日** |

# 第四次实验：分支与循环结构程序设计

学号: 姓名:

## 一、实验目的：

1、掌握分支、循环结构程序设计方法，重点掌握结合程序流程图设计分支、循环结构入、出口的方法。

2、掌握使用DEBUG调试分支、循环结构的方法。

## 二、实验环境（硬件，软件环境）：

硬件环境：MACOS系统

软件环境：DOSBox v0.74-3-3

## 三、实验内容及要求（算法，程序，步骤，方法）：

1、实验内容

（1）在实验报告中说明所用各类转移指令、循环指令的相关原理。

（2）按照题目要求给出自己设计的程序流程图、源程序清单。

（3）对程序执行结果给予解释，并说明观察执行结果的调试方法。

2、实验要求

给定一个字节型数组，长度为10字节：

VARB DB 14, -25, 66, -17, 78, -33, 46, -96, 71, 80

编写一个完整的源程序，统计数组中奇数和偶数的个数，统计结果分别保存在ODDNUM和EVENNUM单元中。

调试并重点记录：

1. 程序设计原理
2. 程序流程图
3. 源程序
4. 实验结果

3、实验步骤

（1）实验所用各类转移指令与循环指令的相关原理

1）JCXZ：一种单条件转移指令，所判断的条件是CX寄存器中的数据，若（CX）=0则转移到标号地址，若（CX）=1则顺序执行，配合循环控制指令形成计数循环结构。

2）JMP：段内直接转移指令，使程序无条件转移至目标地址，（IP）+DISP=（IP）。

3）JAE：无符号数条件转移指令，以CF，ZF标志位综合判断被减数与减数的大小关系，若A>B则转移到目标地址，A<B则顺序执行。

4）JZ：单标志位转移指令，若ZF=1则转移至目标地址，否则顺序执行

5）LOOP：用来实现循环功能，CX存放循环次数。CPU执行loop指令的时候，先CX=CX-1，然后判断CX中的值，不为零则转至标号处执行程序，如果为零则向下执行。（本源码中未使用，但由于其对CX自减1的功能可以简化程序，文中是手动对CX进行减去）

1. 程序流程图及源程序清单

1）程序设计原理：

数码转换：补码可转换为无符号数后再进行奇偶判断，先判断是否为负数，若为负数则应先用NEG指令求相反数。补码具有保奇偶性，直接判断补码最低位也可确定其奇偶性。奇偶数性质判别：判别无符号数或补码的最低位。

算法：故先判断正负性，转换成补码形式，再使用AND语句判断最后一位数，从而得出奇偶性的条件分支，分别统计直至LEN次。

2）程序框图：

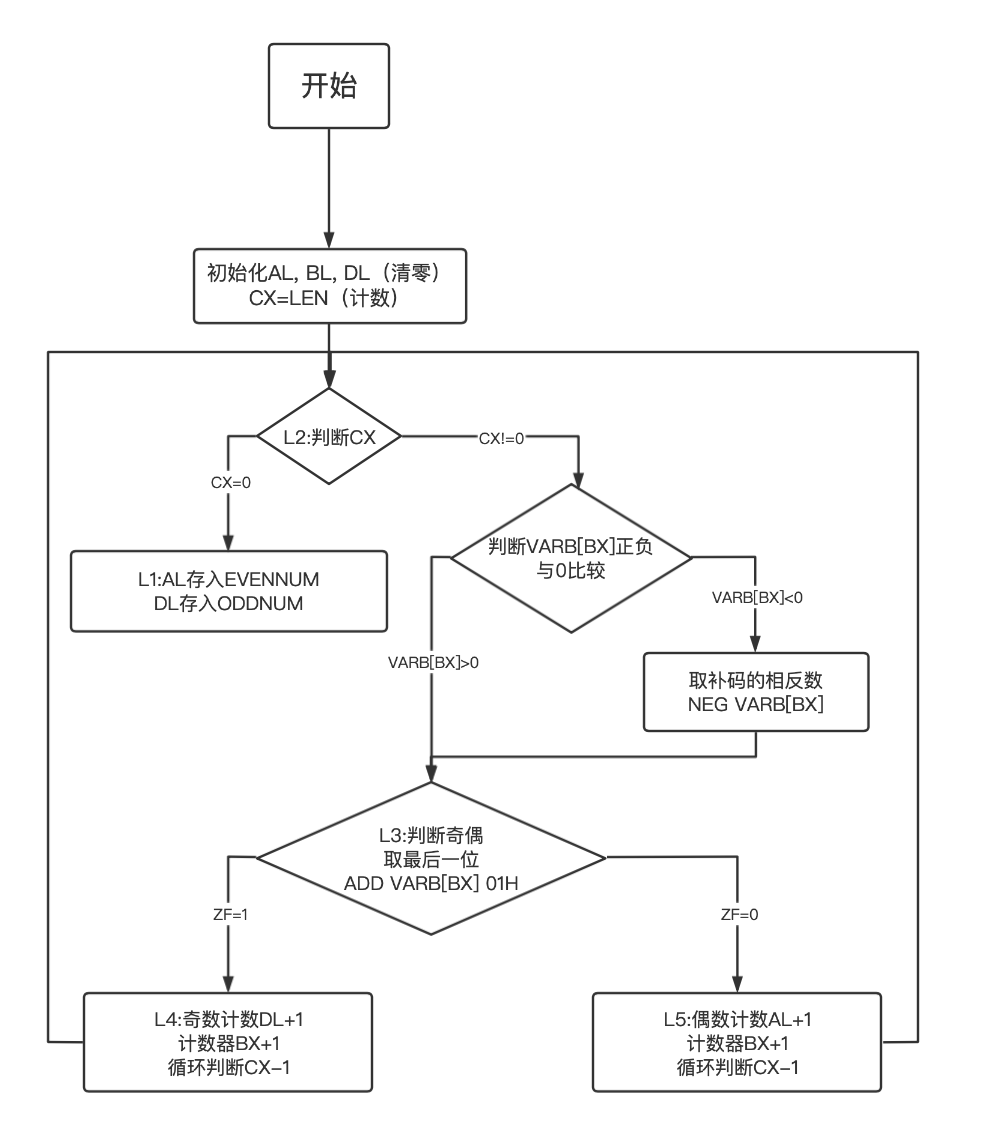


图1.1 程序框图

3）实现源码

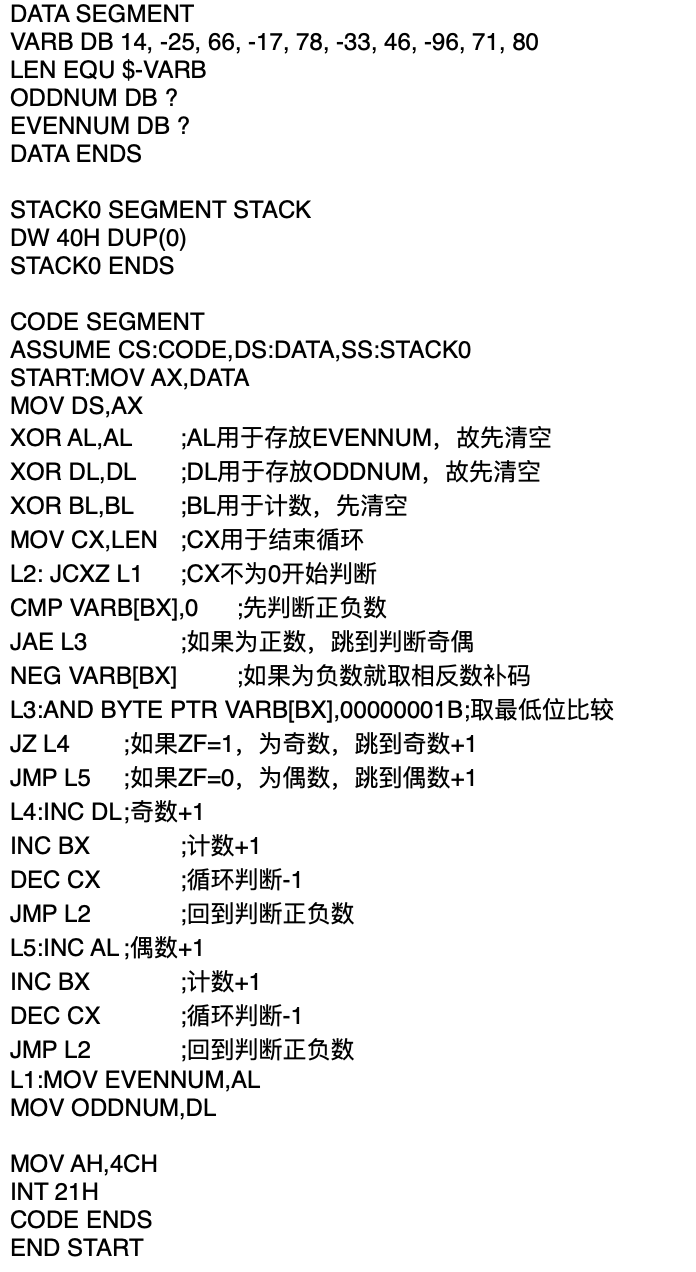


图1.2 程序源码

1. 对程序执行结果给予解释，并说明观察执行结果的调试方法。
2. 下面先说明执行结果的调试方法

A.断点调试命令G——用于断点调试

1.功能：从当前CS、IP指示的位置开始连续执行程序，执行到指定断点处（不包括断点地址指示的指令）暂停执行，在屏幕上显示CPU现场。

2.使用方法：

1. 连续执行，无断点：-g
2. 连续执行到断点位置：-g XXXX:XXXX

3.演示（以-A命令键入测试源代码）

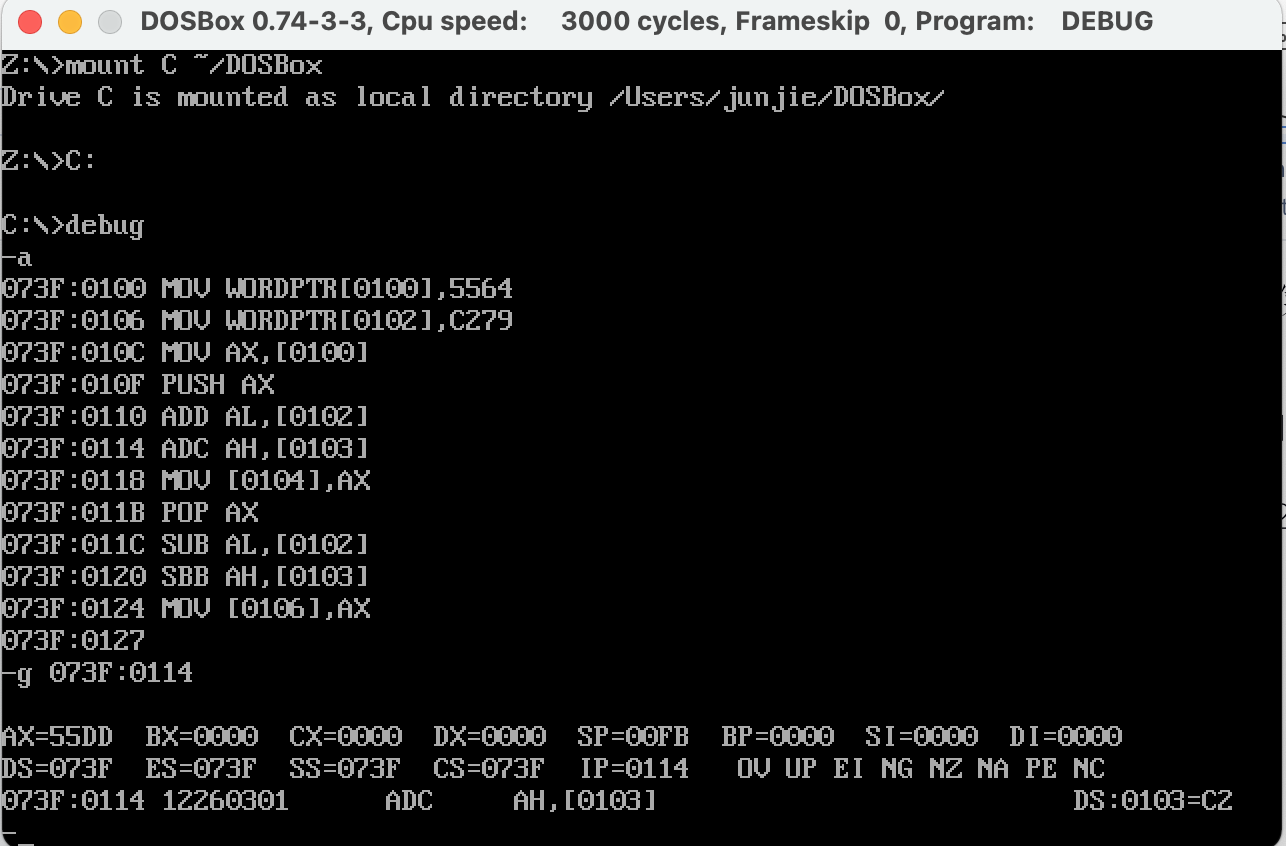


图2.1 调用-g XXXX:XXXX指令（以运行到073F:0114地址为例）

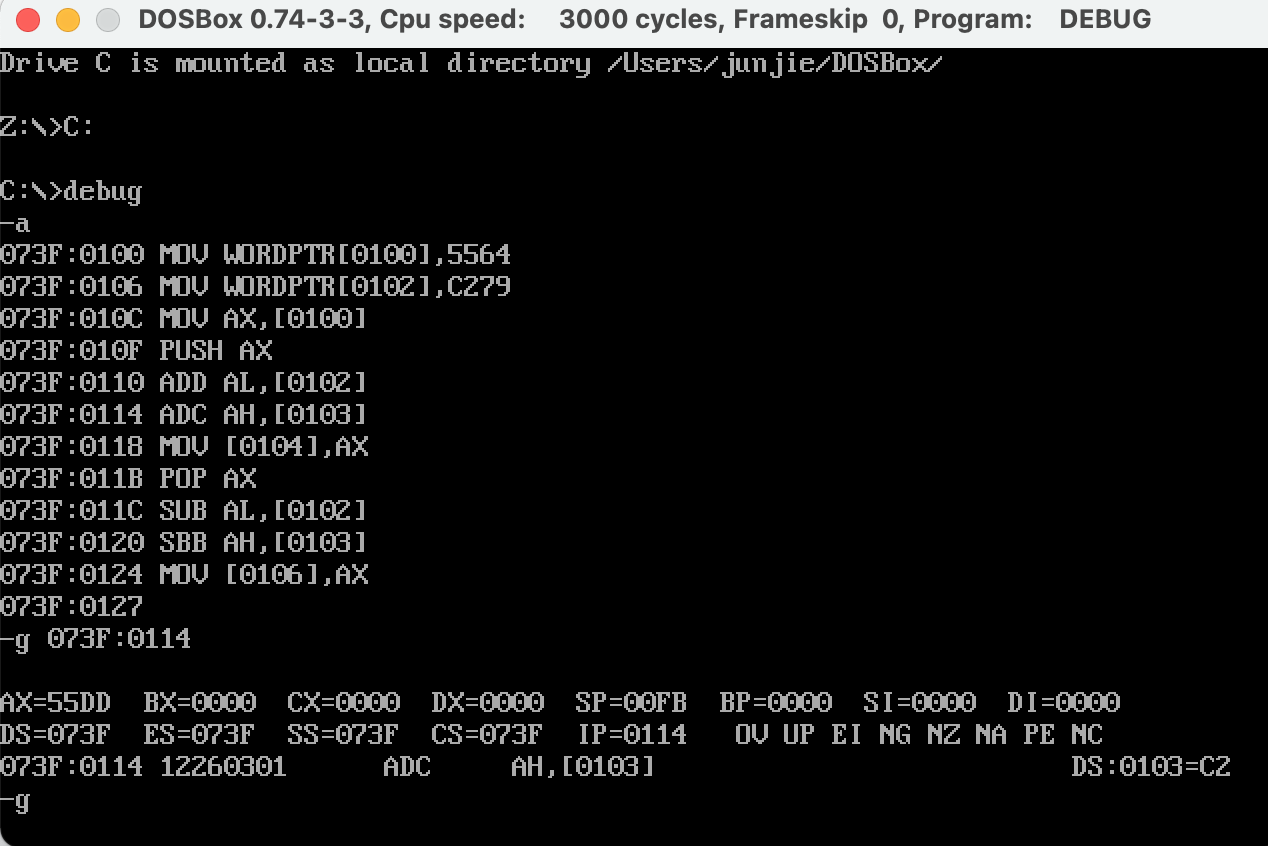


图2.2 调用-g指令（注意：此时程序会一直向下执行，没有终止）

1. 注意：断点位置必须指定为某条指令的首字节地址

B.内存单元查看命令D——用于查看最后变量存储值

1.功能：从当前CS、IP指示的位置开始连续执行程序，执行到指定断点处（不包括断点地址指示的指令）暂停执行，在屏幕上显示CPU现场。

2.使用方法：

1）从默认数据段内存地址开始显示（或接上次所显示的地址继续显示）：-D

2）从指定内存地址开始显示：-D XXXX:XXXX

3.演示（以-A命令键入测试源代码）

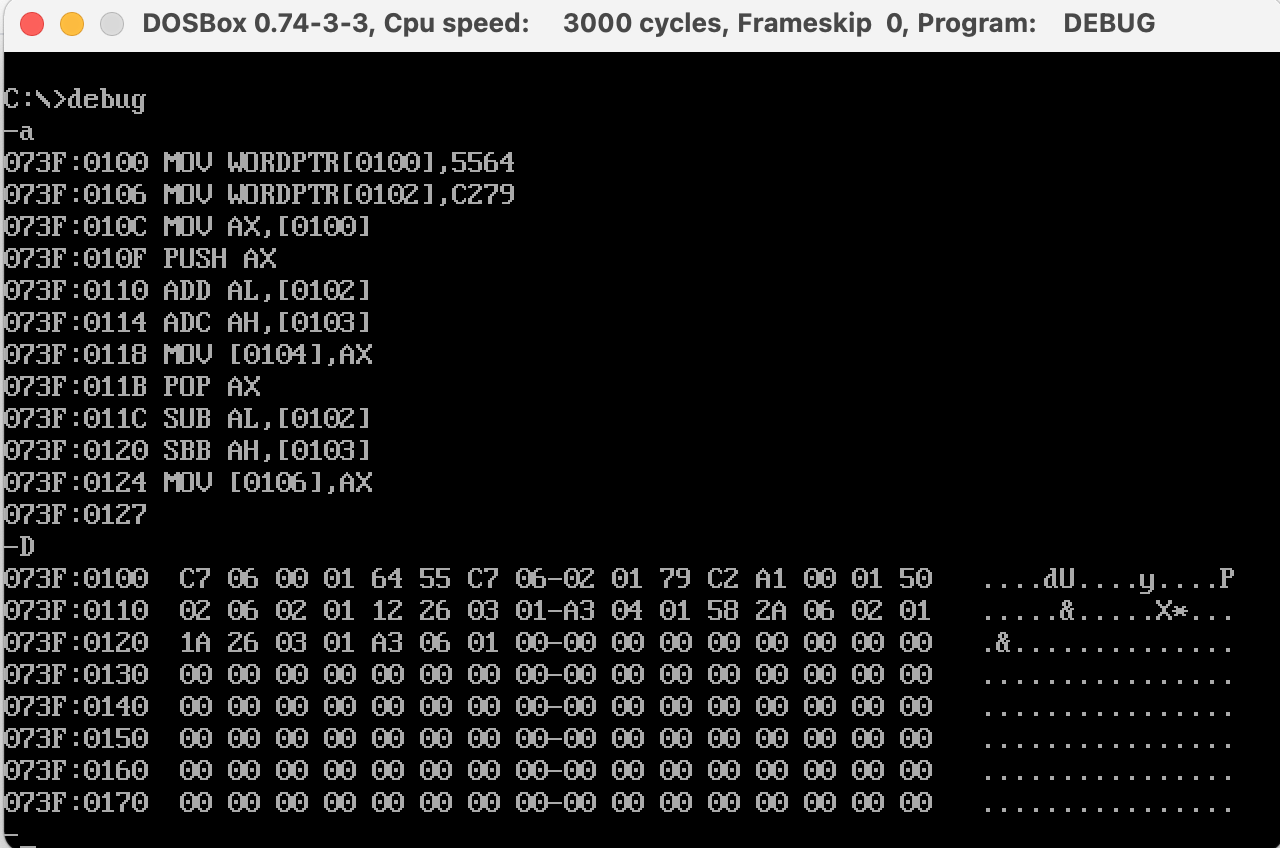


图2.3 调用-d指令

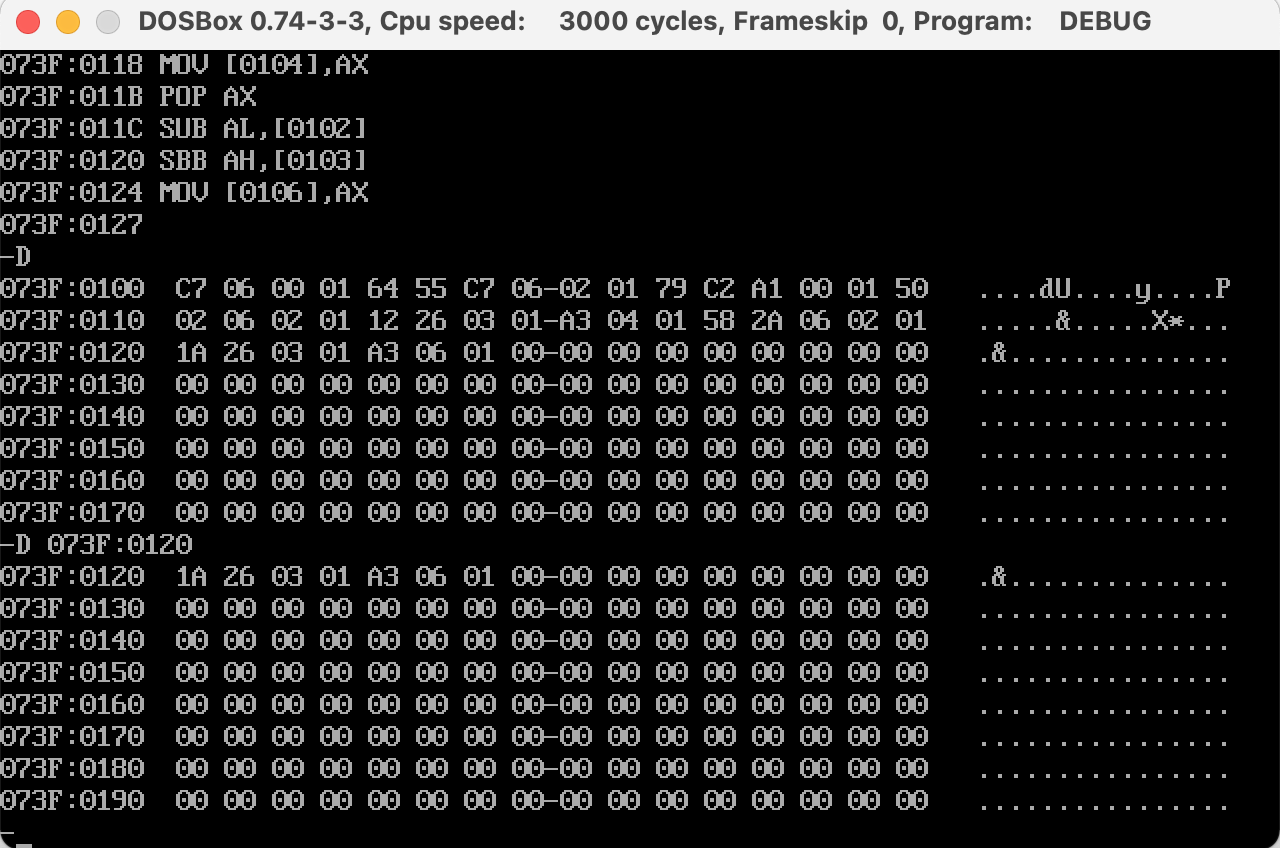


图2.4 调用-d XXXX:XXXX指令（以查看073F:0120内存为例）

1. 注意
2. 从指定位置开始，输出128字节的内容
3. 输出的内容由起始位置，内存单元内容，每个内存单元对应的ASCLL码字符

C.反汇编命令U——用于查看断点地址

将指定逻辑地址起始的机器指令序列反汇编为汇编指令序列，并在屏幕上回显。使用：-u 073D:0107以查看该地址汇编指令，如下所示。

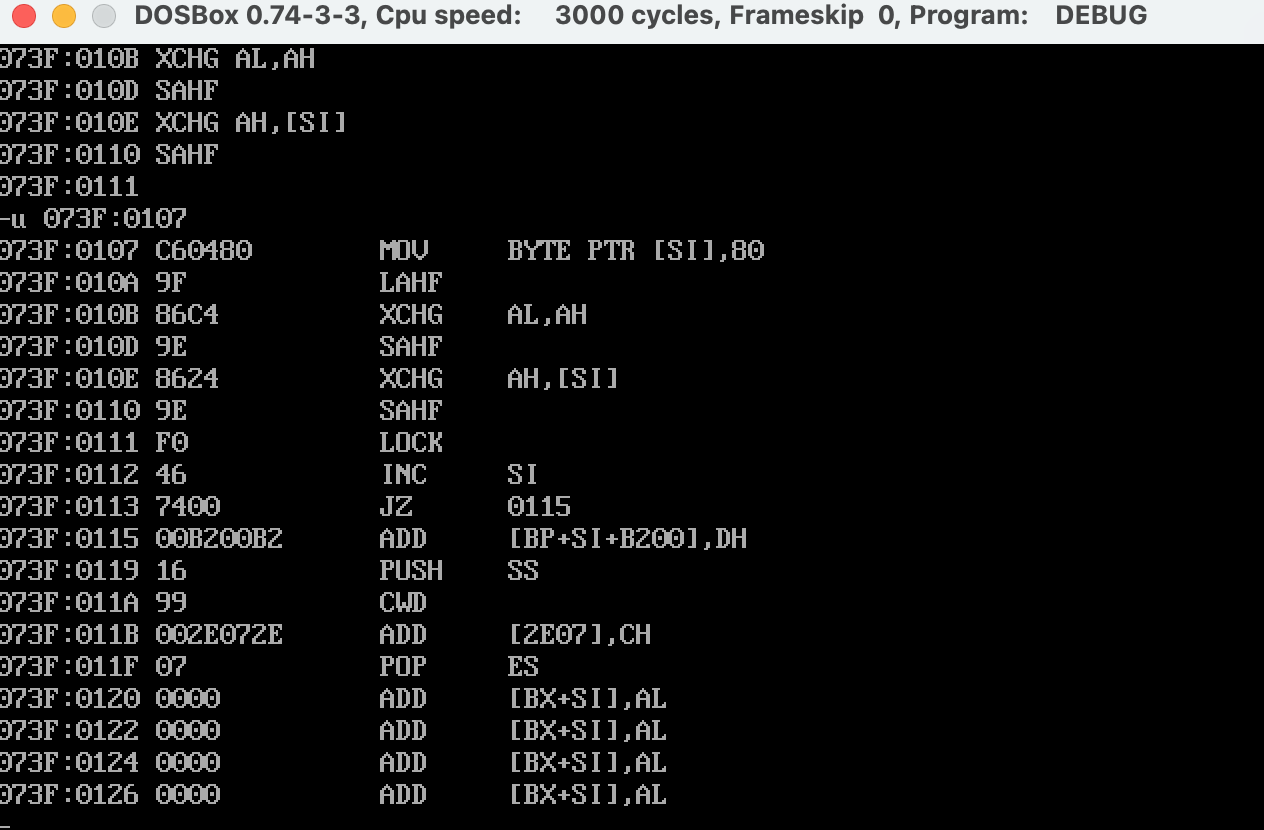


图2.5 汇编指令-u的调用

1. 具体步骤
2. 将源码（见上）保存为.asm格式的文件并放在DOS系统目录下

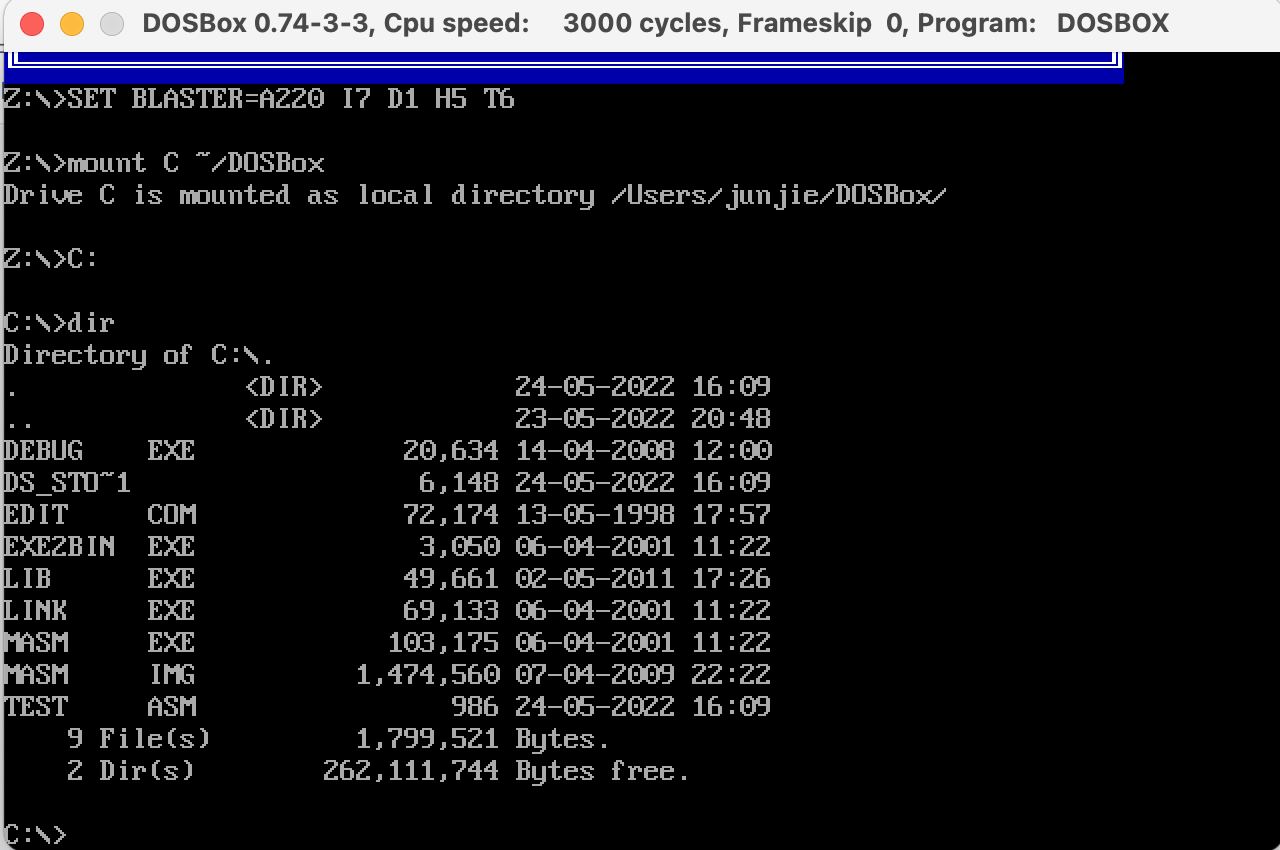


图2.6 dir查看文件

B.使用masm对源文件编译：MASM FILENAME.ASM（文件路径，目标文件文件名，列表文件文件名都作为默认即可，则三个选项回车跳过）

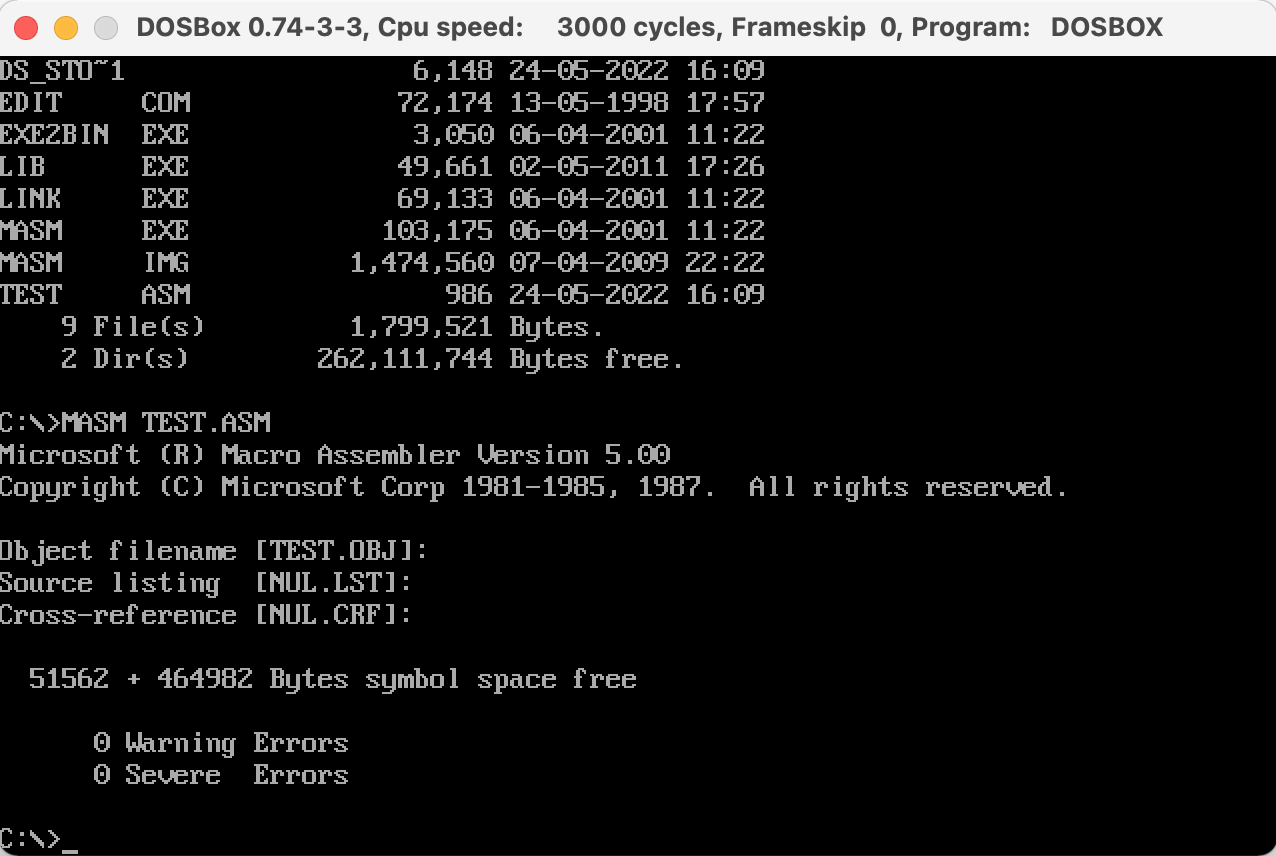


图2.7 编译masm

C.使用link对目标文件链接（文件路径，目标文件文件名，列表文件文件名都作为默认即可，则三个选项回车跳过）

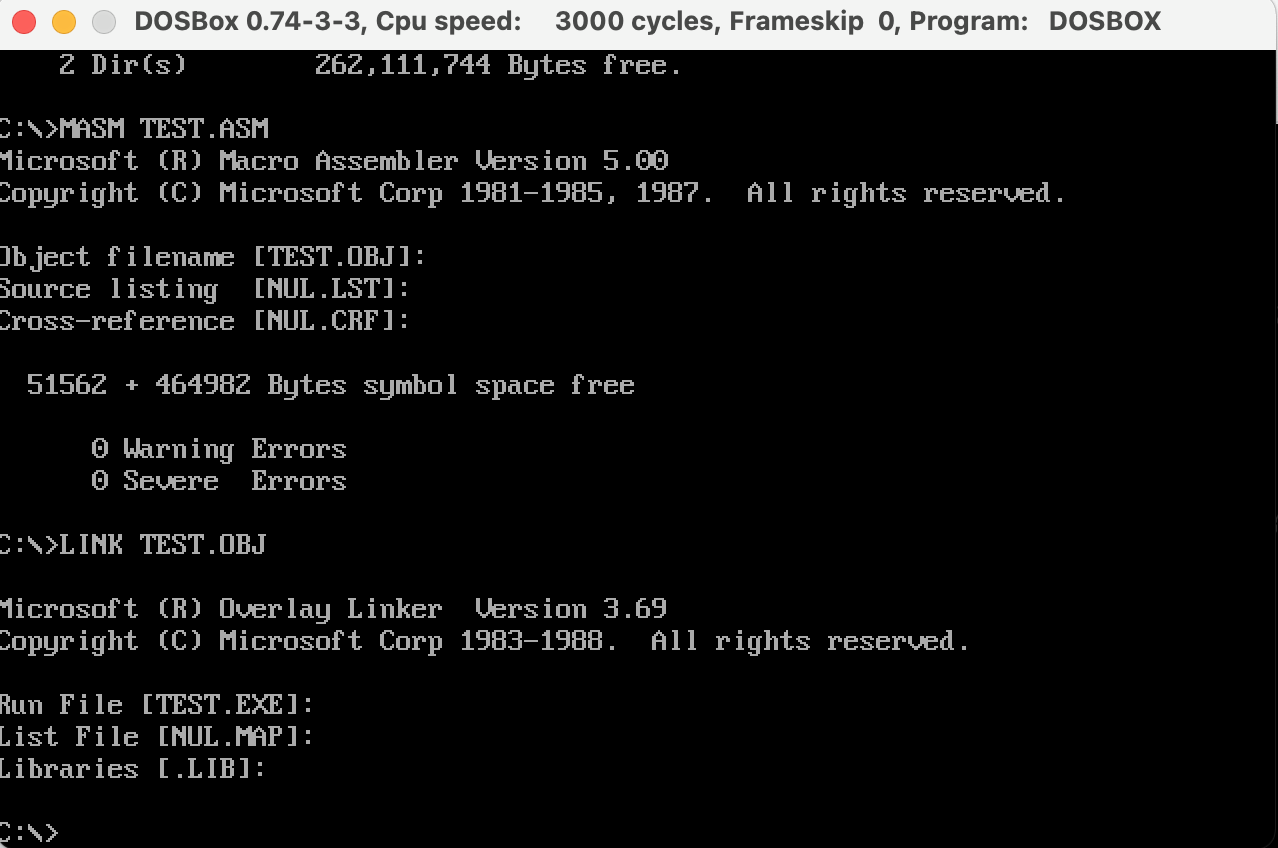


图2.8 链接link

D.首先运行DOSBox，键入-DEBUG TEST.EXE进入debug模式

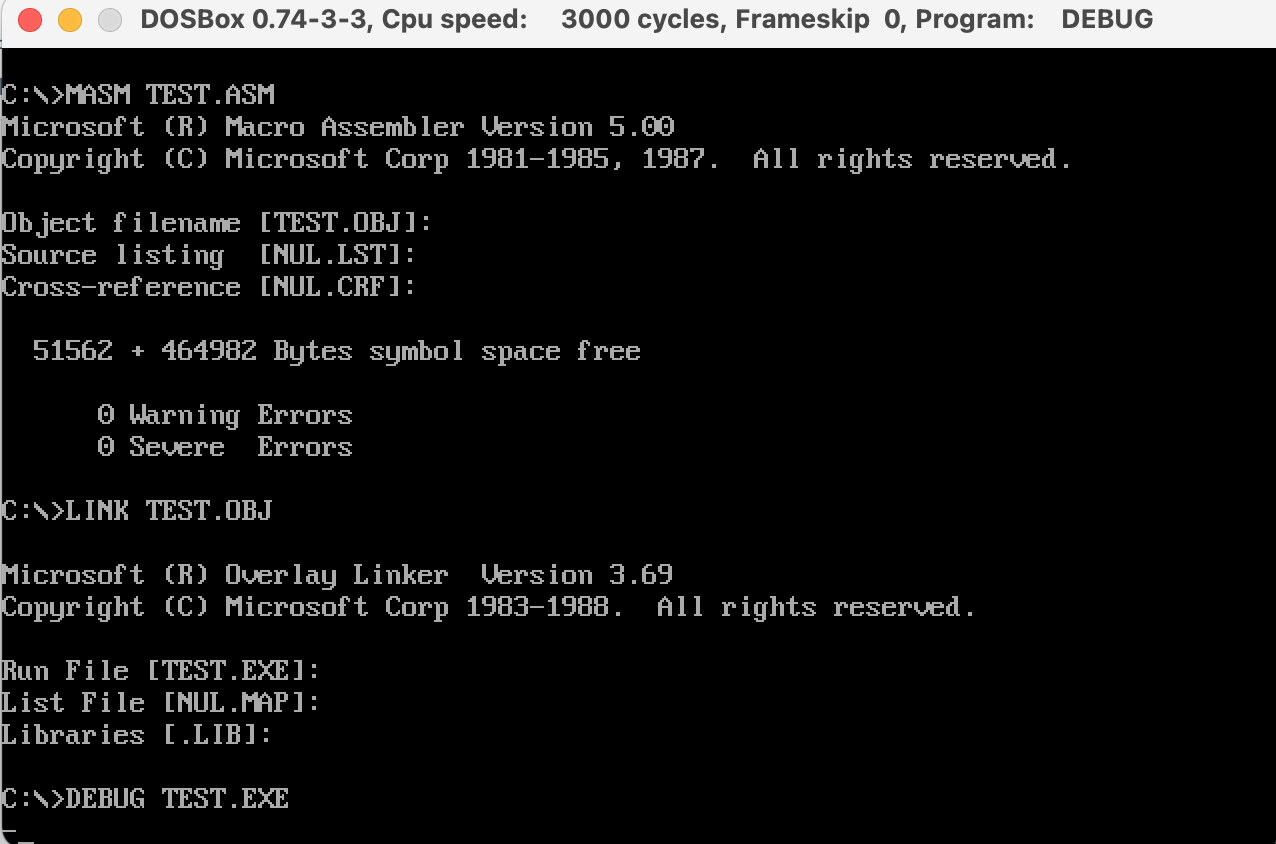


图2.9 进入debug模式

E.使用-u指令查看程序结束指令地址（0073:003C）

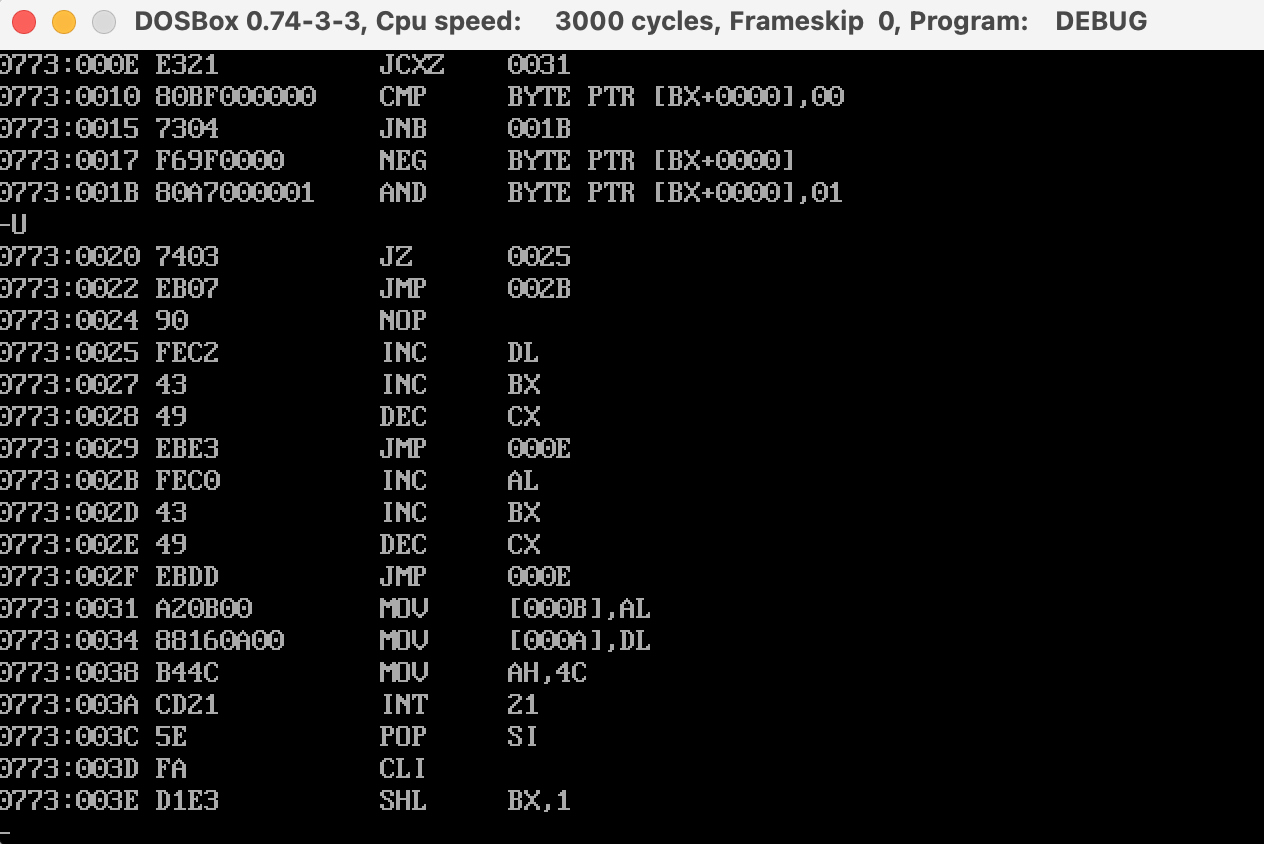


图2.10 查看结束地址

F.使用-g指令设置断点（0073:003C）至程序结束，得到程序结果

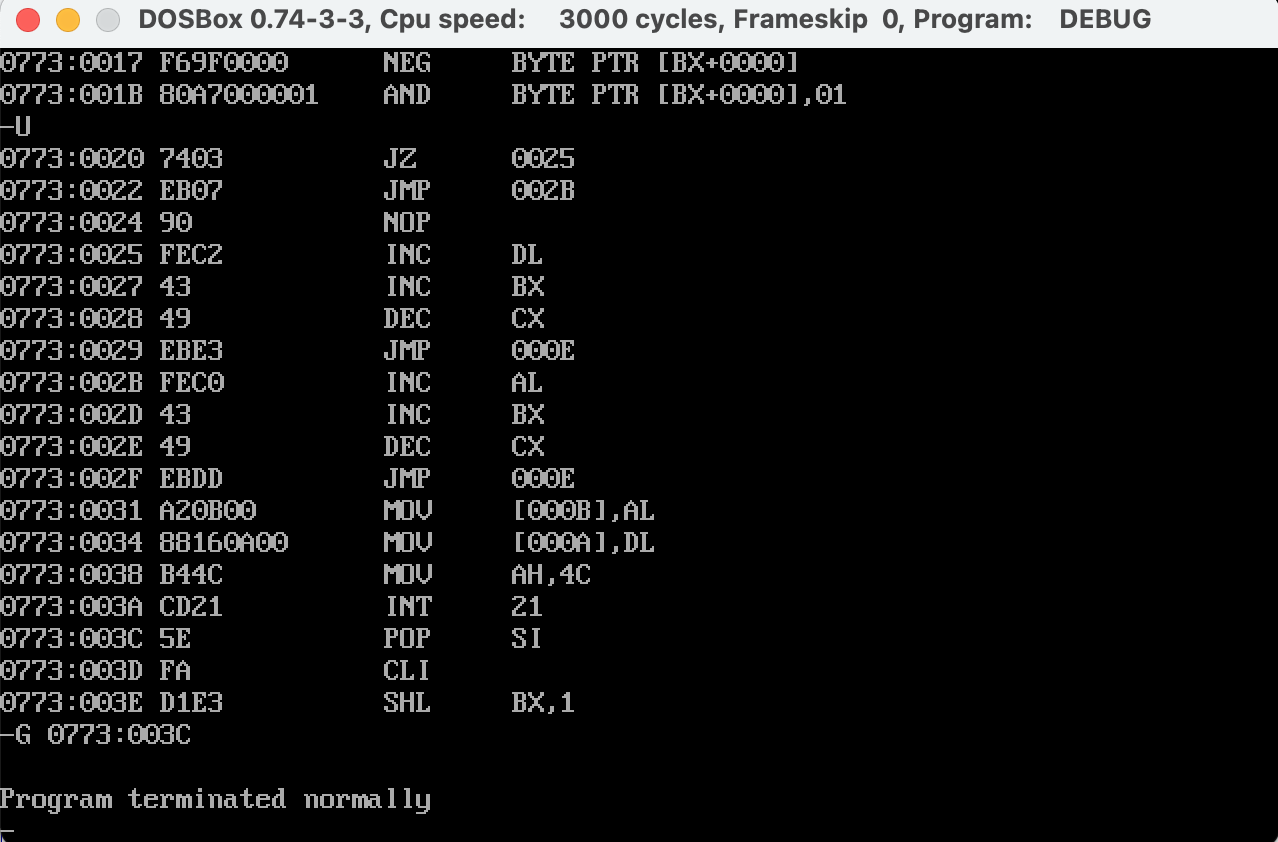


图2.11 程序运行成功

G.使用-r，-d查看实验结果

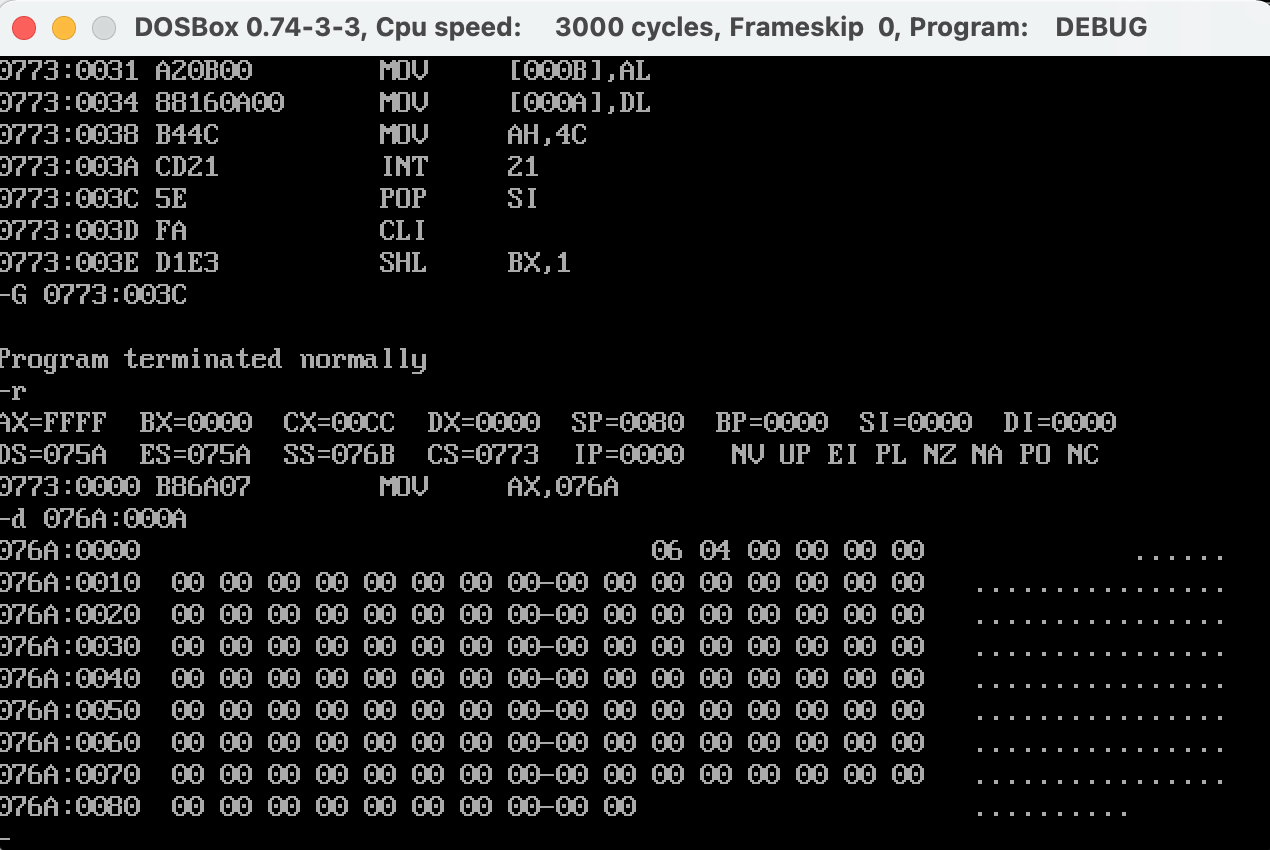


图2.12 程序运行结果

1. 实验源代码

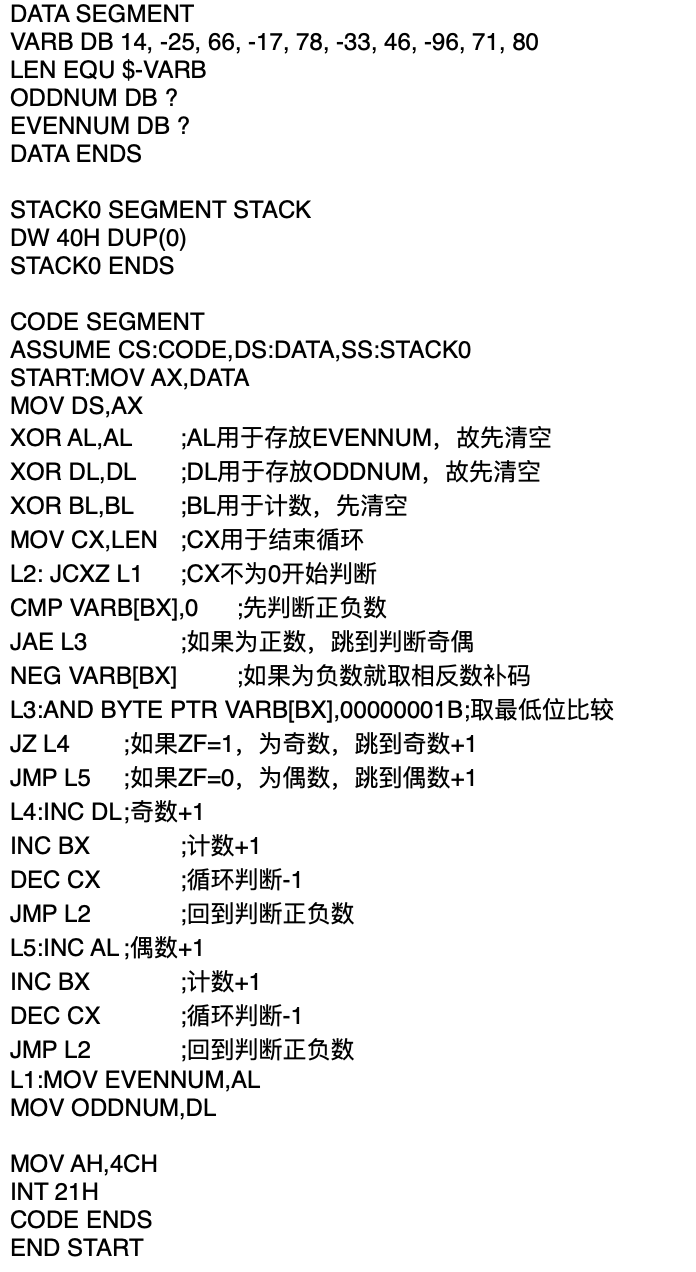


图3 实验源码

## 实验结果分析：

实验结果如下：

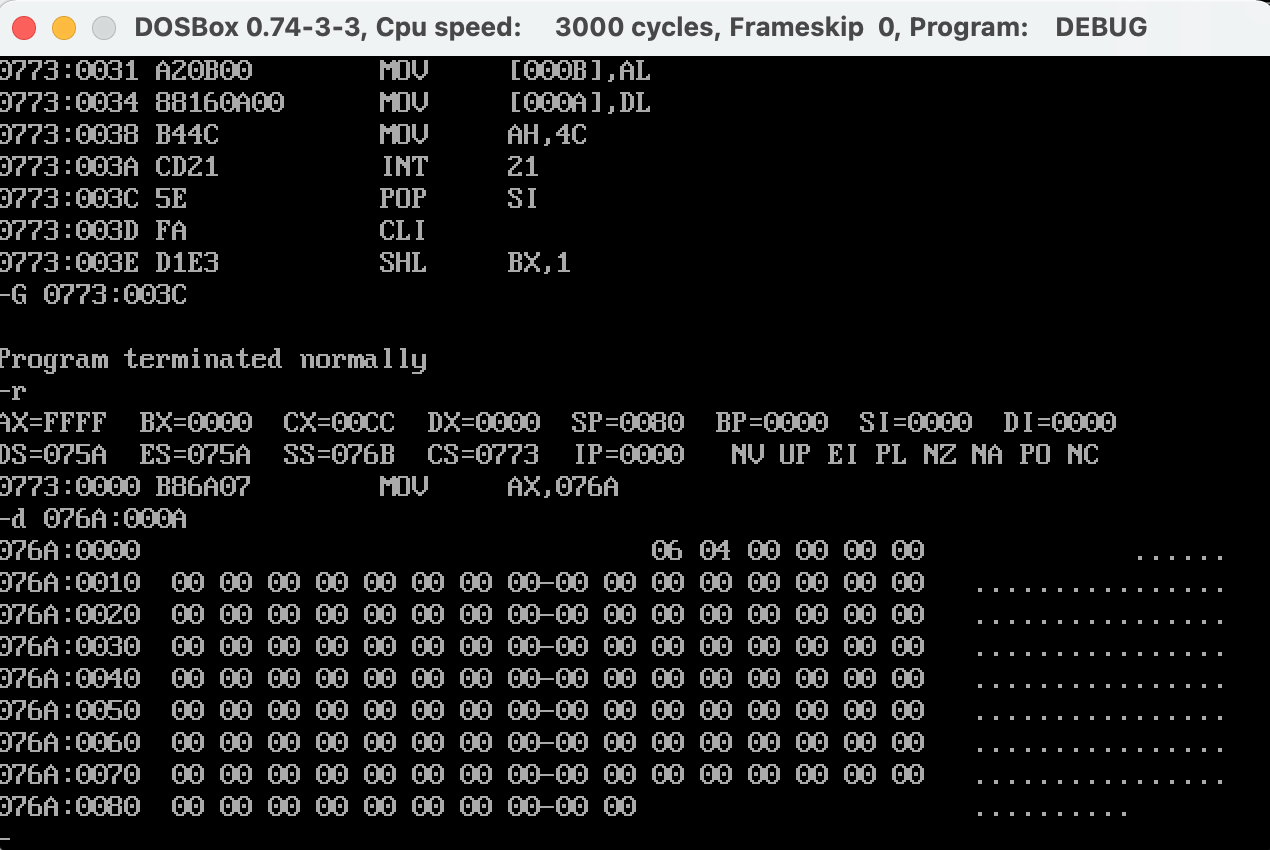


图4 实验结果

下面解释实验结果：

程序执行判断数组中数组的正负性后，将正负数转换成补码形式，从而通过判断末尾1或0即可得出奇偶性，并最终计数存储至ODDNUM与EVENNUM中。

由实验给出十个数字中，有4个奇数，6个偶数，故在DS[000A]为06，DS[000B]为04，即两个变量分别存储的内存位置，则由结果来看实验成功！

## 5、实验体会：

实验过程应注意：

1. 在调试上述程序片段时，应使用r命令先修改DS、SS段寄存器，使之与CS指向不同位置。建议将DS提供的原始段基值加0100H，将SS提供的原始段基值加0200H。避免数据段、堆栈段内的操作覆盖代码段中已录入的指令，影响重复调试。
2. 编写源代码时要大致设计出框图，明确每一步运算结果，编写的程序段必须完整才能运行！
3. 调试时要有耐心，认真分析每一步结果。
4. LOOP指令会让CX自减1。
5. 使用-u可以得到指令地址，从而运用-g设置断点，避免了-t单步调试的麻烦。
6. 查看变量的值是在内存中，用-d查看数据段的段基值加上偏移量即可
7. 通过本次实验，成功设计统计奇偶个数的源程序，实验成功！