

## 实验二(补) 算术运算程序设计

### 一、 实验目的

- 1.进一步掌握顺序程序设计方法。
- 2.熟悉数据传送及算术运算指令的用法。

### 二、 实验内容

- 1.复习教材中顺序结构程序设计的相关内容。
- 2.从 NUM 单元起，存放 a, b, c, d 四个数，其中 a=2, b=-4, c=-18, d=-13，求  $((a*b+10)*10+c)*10+d$  运算，结果存入 RESL 字单元中（设结果小于 16 位数）。
- 3.将修改好的程序 1 输入、汇编、连接并通过 Debug 调试和检查运行结果

### 三、 实验结果（截图）

数据段寄存器 DS 为 076C，故查看 076C:0000 处内存的内容，本次报告中，由于已知结果均小于 16 位数，均使用 16 位进行算术运算。将断点设在给 21 系统功能调用的赋值前，通过寄存器 AX 的值可以观察结果。

```
077D:002E EC      IN      AL,DX
077D:002F 8600      XCHG   AL,[BX+SI]
077D:0031 57      PUSH   DI
077D:0032 56      PUSH   SI
077D:0033 B8BE05      MOV    AX,05BE
077D:0036 50      PUSH   AX
077D:0037 E8C371      CALL   71FD
077D:003A 83C402      ADD    SP,+02
077D:003D 8BF0      MOV    SI,AX
077D:003F 0BF6      OR     SI,SI
-g 29

AX=0007 BX=FFF3 CX=000A DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=076C ES=075C SS=076B CS=077D IP=0029  NU UP EI PL NZ NA PO CY
077D:0029 B8004C      MOV    AX,4C00
-d 0
076C:0000 02 00 FC FF EE FF F3 FF-07 00 00 00 00 00 00 .....
076C:0010 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
076C:0020 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
076C:0030 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
076C:0040 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
076C:0050 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
076C:0060 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
076C:0070 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
```

正确计算结果应为 7，从截图中可以看出 AX 寄存器中的值确为 0007。同时 0007 也正确存入内存单元 DS+8 位置（在此处为 076C:0008 单元）

### 四、 实验报告要求（习题）

- 1.列出源程序（加以注释），说明程序的基本结构。

```
data segment
; 按照题意要求，从 NUM 开始依次存放 a、b、c、d
```

```
    NUM DW 2, -4, -18, -13
; 最后要将结果存入 RESL，此处先声明一下
    RESL DW 0
data ends

; 按照习惯申请一下堆栈
stack segment stack
    DW 128 dup(0)
stack ends

code segment
; 定义段和段寄存器的关系
    assume cs:code, ds:data, ss:stack
start:
; 设置段寄存器
    mov ax, data
    mov ds, ax
; 清空 ax
    xor ax, ax

; 过程中始终保持最终结果存在 ax 中
    mov ax, NUM
    mov bx, NUM+2
; 乘法不能使用立即数，在 cx 中存好常数 10
    mov cx, 0AH
; 完成 a*b
    imul bx
; 完成(a*b)+10
    add ax, 0AH
; 完成((a*b)+10)*10
    imul cx
; 完成((a*b)+10)*10+c
    mov bx, NUM+4
    add ax, bx
; 完成(((a*b)+10)*10+c)*10
    imul cx
; 完成(((a*b)+10)*10+c)*10+d
```

```
    mov bx, NUM+6
    add ax, bx
; 将最终计算结果存入 RESL 中
    mov RESL, ax
; 程序带返回码终止
    mov ax, 4c00h
    int 21h
code ends
end start
```

## 2.说明如何用 Debug 相关命令查看程序 1 的运行结果

下面叙述本次报告使用 debug 查看运行结果的过程

- ① 使用 debug [exe 程序名]，进入程序的调试
- ② 使用 g [断点]运行程序至断点处，断点可以使用相对地址指定，需要注意的是，指定的断点位置的语句并不会运行
- ③ 使用 r 命令观察寄存器，计算过程中把结果存在了 AX 中
- ④ 使用 d [内存地址]查看指定内存单元以后一段的内容，该命令中的地址也可以使用相对地址进行指定，默认查看的是 DS:指定偏移地址的位置

## 3.上机调试过程中遇到的问题是如何解决的

实验过程中，在将数据从数据段 mov 到寄存器时，发现寄存器高 8 位存了对应的数据，不正确。在与同学交流后，也有同样的问题，在互相观察代码后，发现是数据段定义的数据长度与寄存器计算后长度不匹配，导致没有读取足够的字节。在之后的实验中，要注意数据长度的定义与使用。

另外，一开始在使用乘法汇编指令 imul 时，发现乘完之后 AX 结果总是为零，翻看课本后也没找到问题所在。在查阅 8086 汇编手册后发现 imul 后的操作数不能使用立即数，只能使用寄存器或者内存中的内容。

## 五、 个人体会与总结

本次实验，发现出问题的地方往往具有共性，如此，互相交流讨论就显得非常有意义。这次实验中对于数据长度的错误就是在交流中解决的。另外，要学会多查阅官方手册，获取真实的权威指南。