**电工电子实验中心**

**实验报告**

课程名称： 微机原理与接口技术实验

实验名称： 四则运算

姓名： 李荣毅 学号: 161610339

评定成绩： 审阅教师：

实验时间： 2018.10.23

南京航空航天大学

# 实验目的要求

1) 熟悉汇编语言程序的框架结构，掌握顺序结构的编程方法。

2) 熟悉Tddebug 调试环境和 Turbo Debugger 的使用。

3) 理解 X86 内存数据的组织方式。

4) 理解基本的内存寻址方式

# 实验任务

完成32 位无符号数的加法、减法，16 位乘以 16 位，32 位除以16 位除法的四则运算练习。

# 实验代码（写出自己补全的代码，包含适当注释）

(代码及分析)

;加

MOV AX,A

ADD AX,B

MOV C,AX

MOV AX,A+2

ADC AX,B+2

MOV C+2,AX

;减

MOV AX,AD

SUB AX,BD

MOV CD,AX

MOV AX,AD+2

SBB AX,BD+2

MOV C+2,AX

;乘

MOV AX,A1

MUL B1

MOV C1,AX

MOV C1+2,DX

;除

MOV DX,A2+2

MOV AX,A2

DIV B2

MOV C2,AX ;商

MOV C2+2,DX ;余数

RET

# 探究内容（选做）

1. 若需进行有符号数的运算，需要注意什么问题？如何实现？

注意问题：如果进行有符号运算，加法和减法的运算法则和无符号数是一样的，最后得出得的结果也是一样的；而乘法需要用IMUL指令来进行有符号数乘法，而除法需要用IDIV指令来进行有符号数的除法。这些结果最后的读数都以有符号数的方式进行读数。

(代码及分析)

代码：

DATA SEGMENT

A DW 1234H,5678H ;被加数

B DW 0FEDCH,123AH ;加数

C DW 2 DUP (?) ;预留和

AD DW 0FEDCH,0BA98H ;被减数

BD DW 1234H,789AH ;减数

CD DW 2 DUP (?) ;预留差

A1 DW 0D678H ;被乘数

B1 DW 0012H ;乘数

C1 DW 2 DUP (?) ;预留积

A2 DW 5678H,0234H ;被除数

B2 DW 0F234H ;除数

C2 DW 2 DUP (?) ;预留商，余数

DATA ENDS

STACK1 SEGMENT STACK

DB 100 DUP

STACK1 ENDS

CODE SEGMENT

ASSUME CS:CODE,DS:DATA:SS:STACK1

START PROC FAR

PUSH DS

MOV AX,0

PUSH AX

MOV AX,DATA

MOV DS,AX

;加

MOV AX,A

ADD AX,B

MOV C,AX

MOV AX,A+2

ADC AX,B+2 ;ADC

MOV C+2,AX

;减

MOV AX,AD

SUB AX,BD

MOV CD,AX

MOV AX,AD+2

SBB AX,BD+2 ;SBB

MOV CD+2,AX

;乘

MOV AX,A1

IMUL B1

MOV C1,AX

MOV C1+2,DX

;除

MOV DX,A2+2

MOV AX,A2

IDIV B2

MOV C2,AX ;商

MOV C2+2,DX ;余数

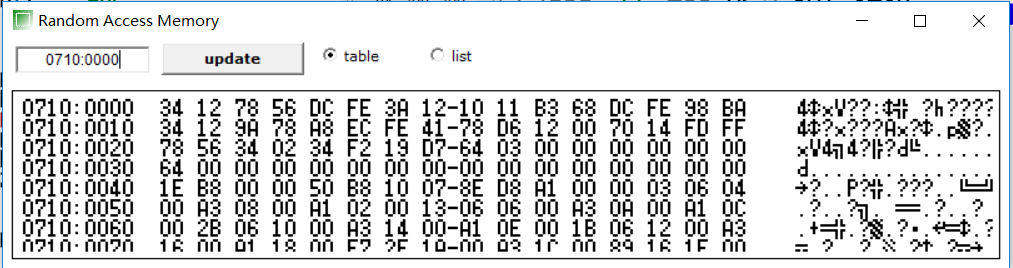
RET

START ENDP

CODE ENDS

END START

分析：



下面的都是有符号数

和：68B31110H

差：41FEECA8H

积：FFFD1470H

商：D719 余数：0364

2) 上述实验中，我们在80X86 的实模式中实现了 32 位的四则运算。例如，乘法运算为16

位乘以 16 位，运算结果为 32 位。请思考如何利用 32 位指令，实现64 位的四则运算？

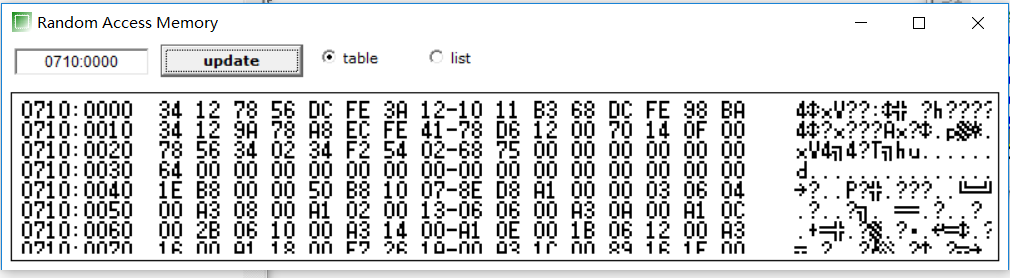
分析：将上述程序中的AX，变成EAX，以此类推，将其他的寄存器多个E变成32位的，泽可以实现64位的四则运算

# 实验的运行数据及分析

1. 实验数据记录

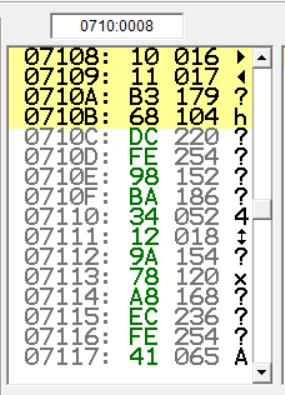
(包括内存数据表及运行结果截图)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DS:0000 | 34 | 12 | 78 | 56 | DC | FE | 3A | 12 |
| DS:0008 | 10 | 11 | B3 | 68 | DC | FE | 98 | BA |
| DS:0010 | 34 | 12 | 9A | 78 | A8 | EC | FE | 41 |
| DS:0018 | 78 | D6 | 12 | 00 | 70 | 14 | 0F | 00 |
| DS:0020 | 78 | 56 | 34 | 02 | 34 | F2 | 54 | 02 |
| DS:0028 | 68 | 75 | 00 | 00 | 00 | 00 | 00 | 00 |

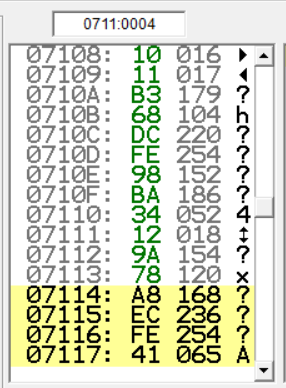


黄色的部分为结果

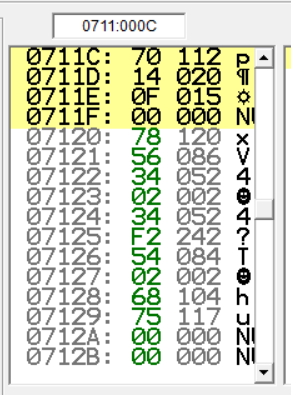
C的结果：



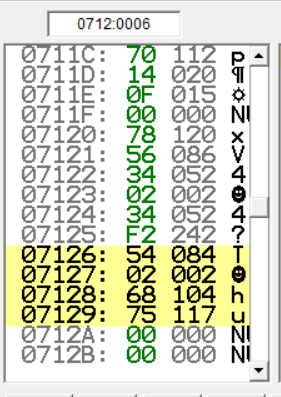
CD的结果：



C1的结果



C2的结果：



1. 数据分析：

（结果分析）

和：68B31110H

该结果为 56781234H+123AFEDCH的和

差：41FEECA8H

该结果为 BA98FEDCH-789A1234H的差

积：000F1470H

该结果为D678H\*12H(无符号数)的积，计算过程中积存在DX:AX中

商：0254H 余数：7568H

该结果为2345678H/F234H（无符号数）的商，被除数存在DX:AX中，商存在AX中，而余数存在在DX中

# 实验讨论及心得体会

（对实验过程、方案选择、设计调整、实验结果等进行讨论分析）

根据已给出的加法指令，来类推出减法指令，低位的数值直接用SUB相减，高位的数值用SBB来进行相减，SBB用考虑CF位，考虑借位的减法。乘法指令将被成数存储在AX中，然后用MUL来进行相乘，16位乘以16位最多有32位，所以最终的结果存储在DX:AX中。除法指令将被除数存储在DX:AX，然后用DIV进行除法，最终的结果商存放在AX中，而余数存放在DX中。无符号数和有符号数的运算区别，加减法的过程是一样的，只是在最终读数的时候是以有符号或者无符号的方式读取；而乘除法有无符号的差别很大，对指令而言，无符号用MUL和DIV，最终的结果是无符号数，而有符号用IMUL和IDIV，最终的结果是有符号数。

建议：可以最终用一个四个运算都混合一起用的表达式来进行四则运算的理解