苏州大学实验报告

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 院、系 | 计算机学院 | | 年级专业 | | 17计科 | | 姓名 | 魏军杰 | 学号 | 1727405175 |
| 课程名称 | | 汇编语言程序设计 | | | | | | | 成绩 |  |
| 指导教师 | | 胡沁涵 | | 同组实验者 | | 无 | | 实验日期 |  | |

|  |  |
| --- | --- |
| 实 验 名 称 | 实验一 嵌入汇编调试、IA-32寄存器和寻址方式的使用 |

1. 实验目的
2. 掌握VC2010嵌入汇编调试；
3. 了解IA32处理器的基本功能；
4. 了解和掌握通用寄存器相关指令；
5. 了解标志寄存器相关概念；
6. 了解段寄存器的概念；
7. 了解和掌握寻址方式；
8. 了解和掌握指令指针寄存器和简单控制转移指令的使用；
9. 了解堆栈和相关指令的使用。
10. 实验内容
11. 在VC2010环境下，编辑运行如下控制台程序。利用VS设置断点进行调试，查看寄存器信息、内存单元数据。下划线处请填写对应寄存器的值，并附上对应指令运行后通用寄存器和标志寄存器的截图。

#include <stdio.h>

int main( )

{

int x=0xFFFF , y=0;

\_asm {

MOV EAX, 12345678H ; EAX=\_\_\_\_\_\_\_\_

MOV ECX, x

ADD AX, CX ; EAX=\_\_\_\_\_\_\_\_

LEA EDX, y ; EDX=\_\_\_\_\_\_\_\_

MOV [EDX], EAX

}

printf("y=%x", y);

return 0;

}

请给出运行结果\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2、在VC2010环境下，使用嵌入汇编的方式实现下列程序中下划线处语句的功能。编辑运行下列程序，然后利用VS反汇编功能，查看对应程序的反汇编结果。比较所写的嵌入汇编指令和反汇编得到的指令有何异同。

#include <stdio.h>

int main( )

{

int x=1,y=2,z=0;

**z=2\*x+5\*y+100;**

return 0;

}

3、请使用VS嵌入汇编形式编写ex21程序。由用户从键盘输入4个字符，存放到字符数组中；把这4个字节数据作为无符号数，分别拼接成2个字数据，还拼接成1个双字数据；显示输出这2个字数据的值和1个双字数据的值。

4、请使用VS嵌入汇编形式编写ex22程序。要求由用户从键盘输入2个整数（整型）；计算这2个整数之差；显示输出上述差的绝对值。

5、假设有一个字符型数组，含有10个字节（8位）整数。可以在定义字符型数组时初始化好这10个数据；也可以由用户输入。请编写程序ex23实现：将这些整数作为有符号数，统计这些整数中正数和负数的个数，并显示输出统计结果。

6、请使用VS嵌入汇编形式编写ex24程序。由用户从键盘输入一个字符串；统计该字符串的长度；显示输出字符串长度。

注意：在上述3、4、5和6题的程序中，输入和输出可以采用高级语言语句，数据处理部分应当采用嵌入汇编。

1. 实验步骤和结果
2. 在VC2010环境下，编辑运行如下控制台程序。利用VS设置断点进行调试，查看寄存器信息、内存单元数据。下划线处请填写对应寄存器的值，并附上对应指令运行后通用寄存器和标志寄存器的截图。

#include <stdio.h>

int main( )

{

int x=0xFFFF , y=0;

\_asm {

MOV EAX, 12345678H ;EAX=0x12345678

MOV ECX, x

ADD AX, CX ;EAX=0x12345677

LEA EDX, y ;EDX=0x002CFB5C

MOV [EDX], EAX

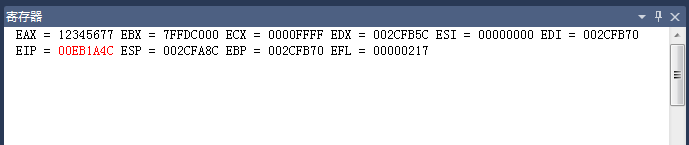
}

printf("y=%x", y);

return 0;

}

请给出运行结果\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



1. 在VC2010环境下，使用嵌入汇编的方式实现下列程序中下划线处语句的功能。编辑运行下列程序，然后利用VS反汇编功能，查看对应程序的反汇编结果。比较所写的嵌入汇编指令和反汇编得到的指令有何异同。

#include <stdio.h>

int main( )

{

int x=1,y=2,z=0;

**z=2\*x+5\*y+100;**

return 0;

}

嵌入的汇编指令：

\_asm{

MOV EAX, x

MOV EBX, y

IMUL EAX, EAX,2

IMUL EBX, EBX,5

ADD EAX, EBX

ADD EAX, 100

MOV [z], EAX

}

反汇编得到的指令：

mov eax,dword ptr [y]

imul eax,eax,5

mov ecx,dword ptr [x]

lea edx,[eax+ecx\*2+64h]

mov dword ptr [z],edx

1. 请使用VS嵌入汇编形式编写ex21程序。由用户从键盘输入4个字符，存放到字符数组中；把这4个字节数据作为无符号数，分别拼接成2个字数据，还拼接成1个双字数据；显示输出这2个字数据的值和1个双字数据的值。

#include <iostream>

using namespace std;

int main()

{

char s[5];

cin >> s;

short a, b;

int d;

\_asm {

MOV AL, s[0]

SAL EAX, 8

MOV AL, s[1]

MOV a, AX

SAL EAX, 8

MOV AL, s[2]

SAL EAX, 8

MOV AL, s[3]

MOV b, AX

MOV [d],EAX

}

cout << a << endl;

cout << b << endl;

cout << d << endl;

return 0;

}

输入：1234

输出为：





1. 请使用VS嵌入汇编形式编写ex22程序。要求由用户从键盘输入2个整数（整型）；计算这2个整数之差；显示输出上述差的绝对值。

#include <iostream>

using namespace std;

int main()

{

int a, b;

cin>>a>>b;

int c;

\_asm {

MOV EAX, a

MOV EBX, b

SUB EAX,EBX

MOV ECX,EAX

SAL ECX,31

CMP ECX,0

NEG EAX

MOV [c],EAX

}

return 0;

}

1. 假设有一个字符型数组，含有10个字节（8位）整数。可以在定义字符型数组时初始化好这10个数据；也可以由用户输入。请编写程序ex23实现：将这些整数作为有符号数，统计这些整数中正数和负数的个数，并显示输出统计结果。

#include <iostream>

using namespace std;

int main()

{

char p[10] = { 2,3,-4,5,6,97,8,-9,-10,-13 };

int pos;

int neg;

\_asm {

MOV EAX, 0;

MOV ESI, 0

MOV CL, 10

LEA EBX, p

NEXT:

MOV DL, [EBX+ESI\*1]

SHR DL,7

ADD AL, DL

INC ESI

DEC CL

JNZ NEXT

MOV pos,EAX

MOV EDX,10

SUB EDX,EAX

MOV [neg],EDX

}

cout << pos << endl;

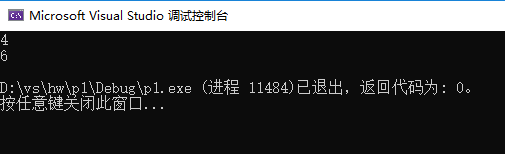
cout << neg << endl;

return 0;

}

输入：char p[10] = { 2,3,-4,5,6,97,8,-9,-10,-13 };

结果：



6、请使用VS嵌入汇编形式编写ex24程序。由用户从键盘输入一个字符串；统计该字符串的长度；显示输出字符串长度。

注意：在上述3、4、5和6题的程序中，输入和输出可以采用高级语言语句，数据处理部分应当采用嵌入汇编。

#include<iostream>

#include<stdio.h>

using namespace std;

int main()

{

char s[] = "123456789";

char\* p = s;

int count;

\_asm {

MOV EAX, p;

MOV ESI,0

MOV ECX,0

NEXT:

MOV cl,[EAX+ESI]

INC ESI

INC CL

DEC CL

JNZ NEXT

DEC ESI

MOV count,ESI

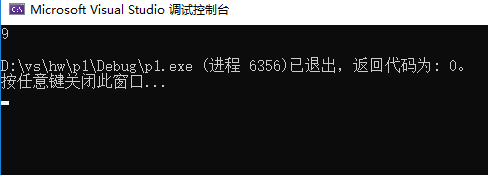
}

cout << count << endl;

return 0;

}

输出：



1. 实验总结

这次实验初步了解了汇编的调试方法以及对程序的底层表示有进一步的了解，学会了汇编的常用的语法，会使用循环和选择语句来实现简单的汇编程序，看到了高级语言看不到的底层的数据表示，进一步提升了我写程序的能力以及debug能力