

# 汇编语言程序设计实验

## 网络\*\*\* \*\*

### 实验一 顺序与分支程序设计

#### 一、实验目的

- 1) 掌握顺序程序设计方法。
- 2) 掌握分支程序的结构及分支程序的设计，调试方法。
- 2) 学习数据传送及算术和逻辑运算指令的用法。
- 3) 熟悉在 pc 机上建立、汇编、连接、调试和运行汇编语言程序的过程。

#### 二、实验内容

##### 1.实验六

从键盘上接收一位十进制数  $x$ ，计算  $y$  值，并以十六进制形式显示出来， $y$  按下列公式计算。

$$Y = \begin{cases} X^2 + X & (X = 3) \\ X^2 - 2X & (X = 4) \\ X^2 & (X = 5) \\ X^2 / 2 & (X = 6) \end{cases}$$

##### 2..实验七

实验内容：从键盘上接收两个一位十六进制数  $x$  和  $y$ ，然后再输入一个  $a-d$  之间的一个字符，按下列要求计算。

- a) 当输入字符为  $a$ ，则计算  $x+y$ ，并以十六进制形式显示出来
- b) 当输入字符为  $b$ ，则计算  $|x-y|$ ，并以十六进制形式显示出来
- c) 当输入字符为  $c$ ，则计算  $x*y$ ，并以十六进制形式显示出来
- d) 当输入字符为  $d$ ，则计算  $x/y$ ，并以十六进制形式显示出来

#### 三、实验代码

实验六

DATE SEGMENT

```

X DB ?
DATE ENDS          ; 数据段
CODE SEGMENT
    ASSUME CS:CODE,DS:DATE
START:MOV AX,DATE          ; 代码段
    MOV DS,AX              ; 初始化 ds 寄存器
    MOV AH,1
    INT 21H                ; 读文件到缓冲区
    CMP AL,33H            ; 比较指令
    JB L0                  ; A<B
    CMP AL,36H
    JA L0
    AND AL,0FH             ; 逻辑与指令
    MOV X,AL               ; al 给 x
    MOV AL,X
    CMP AL,3H
    JE L1                  ; 执行 L1 命令
    CMP AL,4H
    JE L4                  ; 执行 L4 命令
    CMP AL,5H
    JE L6                  ; 执行 L6 命令
    CMP AL,6H
    JE L7                  ; 执行 L7 命令
L1:MUL AL
    ADD AL,X                ; al+x
L5:MOV BL,AL
    MOV CL,4
    SHR AL,CL              ; 移位指令
    CMP AL,9
    JLE L2                 ; 转移指令。小于或等于, 或者不大于则转移
    ADD AL,7
L2:ADD AL,30H              ; 寄存器 al 加 0x30
    MOV DL,AL
    MOV AH,2                ; 把 2 放入 ah 里
    INT 21H
    AND BL,0FH             ; 逻辑与运算指令
    CMP BL,9
    JLE L3
    ADD BL,7                ; 相加
L3:ADD BL,30H
    MOV DL,BL
    MOV AH,2
    INT 21H
    JMP L0

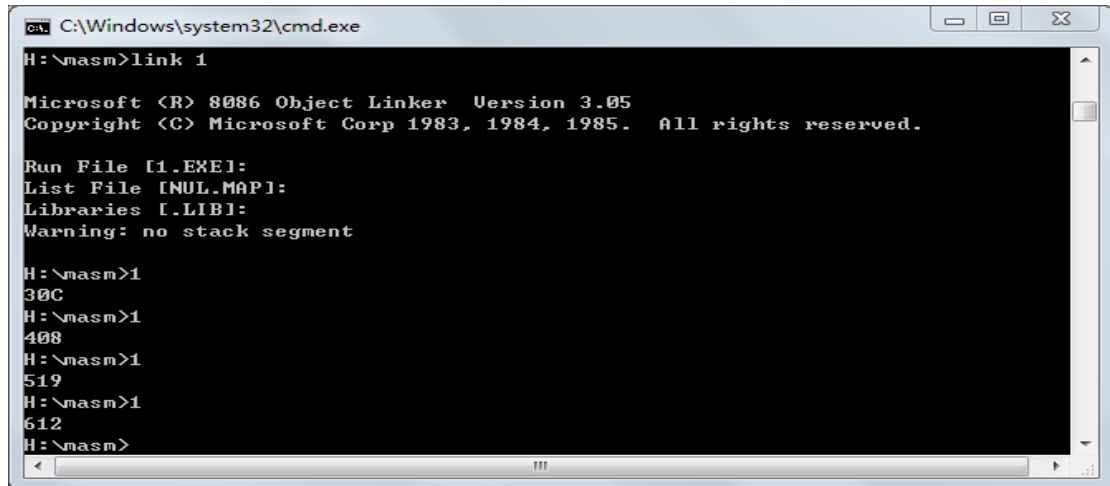
```

```
L4:MUL AL
  SUB AL,X
  SUB AL,X
  JMP L5
L6:MUL AL
  JMP L5
L7:MUL AL
  SHR AL,1
  JMP L5
L0:MOV AH,4CH
  INT 21H
CODE ENDS
END START
实验七
DATA SEGMENT
  X DB 0
  Y DB 0
  W DB 0
DATA ENDS
CODE SEGMENT
ASSUME CS:CODE,DS:DATA
START:MOV AX,DATA
  MOV DS,AX
  XOR AX,AX
  MOV AH,1
  INT 21H
  CMP AL,39H
  JBE L1
  SUB AL,07H
L1:SUB AL,30H
  MOV X,AL
  MOV AH,1
  INT 21H
  CMP AL,39H
  JBE L2
  SUB AL,07H
L2:SUB AL,30H
  MOV Y,AL
  MOV AH,1
  INT 21H
A:CMP AL,41H
  JNE B
  XOR AL,AL
  MOV AL,X
```

```
ADD AL,Y
MOV W,AL
JMP L3
B:CMP AL,42H
JNE C
XOR AL,AL
MOV AL,X
SUB AL,Y
MOV W,AL
CMP AL,0
JG L3
NEG AL
MOV W,AL
JMP L3
C:CMP AL,43H
JNE D
XOR AL,AL
MOV AL,X
MUL Y
MOV W,AL
JMP L3
D:XOR AX,AX
MOV AL,X
DIV Y
MOV W,AL
L3:MOV DL,AL
MOV CL,4
SHR DL,CL
CMP DL,9
JNA L4
ADD DL,07H
L4:ADD DL,30H
MOV AH,02H
INT 21H
MOV DL,W
AND DL,0FH
CMP DL,9
JNA L5
ADD DL,07H
L5:ADD DL,30H
MOV AH,02H
INT 21H
MOV AH,4CH
INT 21H
```

CODE ENDS  
END START

#### 四、实验结果：



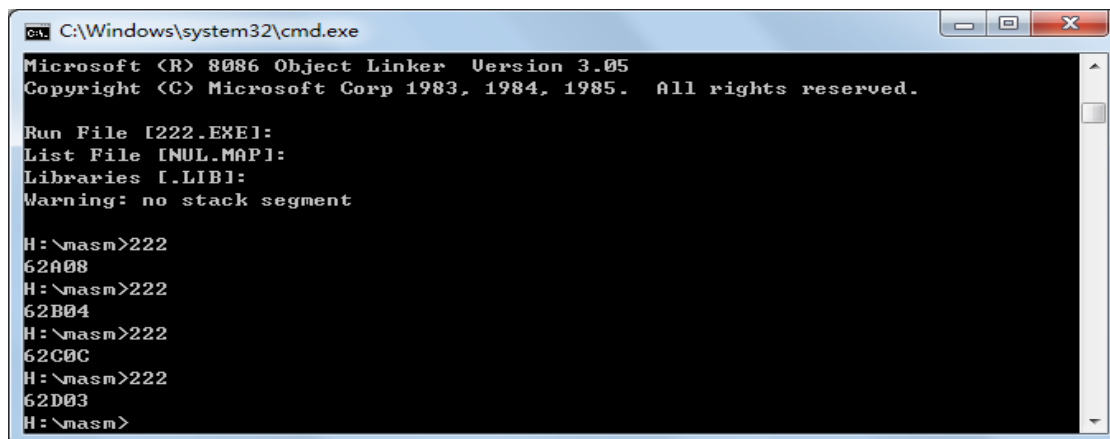
```
C:\Windows\system32\cmd.exe
H:\masm>link 1

Microsoft (R) 8086 Object Linker  Version 3.05
Copyright (C) Microsoft Corp 1983, 1984, 1985.  All rights reserved.

Run File [1.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:
Warning: no stack segment

H:\masm>1
30C
H:\masm>1
408
H:\masm>1
519
H:\masm>1
612
H:\masm>
```

实验六结果



```
C:\Windows\system32\cmd.exe
H:\masm>link 222

Microsoft (R) 8086 Object Linker  Version 3.05
Copyright (C) Microsoft Corp 1983, 1984, 1985.  All rights reserved.

Run File [222.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:
Warning: no stack segment

H:\masm>222
62A08
H:\masm>222
62B04
H:\masm>222
62C0C
H:\masm>222
62D03
H:\masm>
```

实验七结果

#### 五、实验心得：

(1) 通过实验使我知道了和熟悉了汇编程序设计的步骤

1.建立数学模型 遇到一个题目，首先要进行全面的分析，根据问题提出的条件，特点，找出规律，找出问题的数学模型或者制定解决问题的规律。

2.绘制程序流程图 流程图能够把比较抽象的问题变得而形象化，并且有助于理清楚思路，便于程序的编制，在程序的易读性和易维护性上有很大的帮助。

3.合理分配存储空间和寄存器 汇编语言能够直接的调用存储器和寄存器。程序运行时,大两量的数据,中间结果或最后的结果都存放在存储器中,且运行的程序目标代码在存储器中,节约空间是一个好的程序应该注意的问题。

4.编制程序 根据语法指令编制程序

5.人工检查 用人工的方式检查程序是否有错,并及时的改正。

6.上机调试 只有通过上机才能验证程序是否正确,经过反复的运行调试,才能得到正确的程序。

(2) 掌握了分支程序的结构及分支程序的设计,调试方法和数据传送及算术和逻辑运算指令的用法。

# 汇编语言程序设计实验

## 网络\*\*\* \*\*

### 实验二 循环与子程序程序设计

#### 一、 实验目的

- 1) 加深对循环结构的理解。
- 2) 掌握循环程序的设计方法。
- 3) 学习子程序的定义和调用方法。
- 4) 掌握子程序、子程序的嵌套、递归子程序的结构。
- 5) 掌握子程序设计、编制及调试。
- 6) 熟练掌握 DEBUG 的常用命令,学会用 DEBUG 调试程序。

#### 二、 实验内容

- 1) 编制在屏幕上显示九九乘法表的程序。

2) 编制在屏幕上显示用\*组成的三角形的程序。

### 三、实验代码

#### 实验一

```
DATA SEGMENT
S2 DB ?
S1 DB ?
X DB 1
Y DB 1
S DB 0AH
FCH=2AH
FDH=3DH
FKG=20H
FHH=0AH
FHC=0DH
DATA ENDS
STACK1 SEGMENT STACK
DB 20H DUP(0)
STACK1 ENDS
CODE SEGMENT
ASSUME CS:CODE,DS:DATA,SS:STACK1
START:MOV AX,DATA
```

```
MOV DS,AX
MOV CL,9
LL1:
PUSH CX           ;将CL压入堆栈段
MOV CL,Y
LL2:
CALL L1          ;调用子程序L1
INC X
LOOP LL2
MOV X,1
INC Y
MOV DL,FHH      ;输出换行和回车
MOV AH,2
INT 21H
MOV DL,FHC
MOV AH,2
INT 21H
POP CX          ;从堆栈段中取出CL
LOOP LL1
MOV AH,4CH      ;程序结束
INT 21H
L1 PROC        ;计算X*Y=Z并且输出X*Y=Z的子程序
```

```
MOV DL,X
ADD DL,30H
MOV AH,2
INT 21H
MOV DL,FCH
MOV AH,2
INT 21H
MOV DL,Y
ADD DL,30H
MOV AH,2
INT 21H
MOV DL,FDH
MOV AH,2
INT 21H
MOV AL,X
MUL Y
DIV S
MOV S1,AH
ADD AL,30H
MOV DL,AL
MOV AH,2
INT 21H
```

```
ADD S1,30H
MOV DL,S1
MOV AH,2
INT 21H
MOV DL,FKG           ;输出空格
MOV AH,2
INT 21H
RET
L1 ENDP             ;子程序结束
CODE ENDS
END START
```

## 实验二

```
DATA SEGMENT
N DB ?
FCH=2AH
FDH=3DH
FKG=20H
FHH=0AH
FHC=0DH
DATA ENDS
CODES SEGMENT
ASSUME CS:CODES,DS:DATA
```

```
START:
MOV AX,DATA
MOV DS,AX
MOV AH,1           ;输入一位16进制数确定输出行数 INT 21H CMP AL,39H JBE L1
SUB AL,07H
L1:SUB AL,30H MOV N,AL
XOR BX,BX
MOV DL,FCH
MOV CL,N
MOV DL,FHH        ;输出换行和回车
MOV AH,2
INT 21H
MOV DL,FHC
MOV AH,2
INT 21H
LOP:PUSH CX
INC BL
MOV CL,BL
MOV DL,FCH
LOP2:MOV AH,02H
INT 21H
LOOP LOP2
```

```
POP CX
MOV DL,0AH
MOV AH,02H
INT 21H
MOV DL,FHC
MOV AH,02H
INT 21H
CMP BL,9
JZ L2
LOOP LOP
L2:MOV AH,4CH      ;程序结束
INT 21H
CODES ENDS
END START
```

四、实验结果：

```
命令提示符
  0 Warning Errors
  0 Severe Errors

E:\masm>LINK 2

Microsoft (R) 8086 Object Linker Version 3.05
Copyright (C) Microsoft Corp 1983, 1984, 1985. All rights reserved.

Run File [2.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:

E:\masm>2
1*1=01
1*2=02 2*2=04
1*3=03 2*3=06 3*3=09
1*4=04 2*4=08 3*4=12 4*4=16
1*5=05 2*5=10 3*5=15 4*5=20 5*5=25
1*6=06 2*6=12 3*6=18 4*6=24 5*6=30 6*6=36
1*7=07 2*7=14 3*7=21 4*7=28 5*7=35 6*7=42 7*7=49
1*8=08 2*8=16 3*8=24 4*8=32 5*8=40 6*8=48 7*8=56 8*8=64
1*9=09 2*9=18 3*9=27 4*9=36 5*9=45 6*9=54 7*9=63 8*9=72 9*9=81

E:\masm>
```

实验一结果

```
C:\WINDOWS\system32\cmd.exe

3
**
***

E:\masm>3
5
**
***
****
*****

E:\masm>3
8
**
***
****
*****
*****
*****
*****

E:\masm>
```

实验二结果

## 五、实验心得

五、循环程序由初始化部分、循环部分、调整部分、控制部分组成

1) 两种方式，一种是先判断在执行，另一种是先执行再判断。

2) 两种控制循环的方式

a) 计数控制循环

一般使用 LOOP 指令，要先把循环的次数放入 CX

中，在多重循环的计数控制中要注保存和复原上一次循环的断点信息，在保留和复原时可以使用堆栈进行处理。

自认为可以理解为一根盘绕了确定的圈数的绳子，又进入的头，又出去的尾。

#### b) 条件控制循环

一般使用 **JMP** 指令使程序跳到这一段的开头，用 **CMP** 在循环中进行比较判断，再根据程序要求，选择适当的转移条件，跳出循环。

自认为可以理解为一圆形的导轨，并在导轨上设置一个初口，在符合条件时就在导轨上脱离，否则，将会在导轨上一转下去。

3) 再循环中可以同时使用两者，当即已知循环的次数限制范围，但又要求在符合条件时结束循环是，就要同时使用两者。

六、子程序的使用大大的简化了程序设计的思路，在完成一个程序时，可以简单的把一个程序划分为几个完整的几部分，使每一部分独立的完成特定的一部分和计算任务。

子程序的使用还可以大大的减小程序的体积，在一程序中，如果需要多次反复的是用某一段代码的话，就可以将这段代码设定为子程序，以后想使用时，只要一 **CALL** 再一 **RET** 就 OK 啦。

我认为使用子程序的关键在于明白子程序的机理，明白子程序是如何保存和恢复断点的信息的，只有真正的理解了这一点才能更好地利用好子程序。

子程序的参数传递的方式有很多种，但是在堆栈段传递参数时，应该注意使用带立即数的返回的方式，在利用寄存器 **BP** 访问堆栈段来取得参数。

# 汇编语言程序设计实验

## 网络\*\*\* \*\*

### 实验三 算术运算与代码转换程序设计

#### 一、实验目的

- 1) 掌握算术运算程序的设计方法。
- 2) 掌握代码转换程序的设计方法。

- 3) 进一步掌握各种程序结构。
- 4) 熟练掌握和使用用 DEBUG 调试程序。

## 二、实验内容

- 1) 编制两个多字节整数加法和减法程序，并将结果按十六进制形式显示在屏幕上。
- 2) 编写程序计算从键盘输入两个不超过四位的十进制数的和、差、积、商，并以十进制形式输出。
- 3) 编写程序把从键盘输入的以回车为结束符的字符串中的空格去掉并按相反的顺序输出。

## 三 实验代码、

### 实验 1

```
DATA SEGMENT
NUM1 DB 22H,33H,44H,55H,66H,77H
NUM1L DB 6H
NUM2 DB 22H,22H,22H,22H
NUM2L DB 4H
OUTCOME1 DB 10H DUP(0)
OUTCOME2 DB 10H DUP(0)
N DB 0H
DATA ENDS
```

```
STACK1 SEGMENT PARA STACK
DW 20H DUP(0)
STACK1 ENDS
CODE SEGMENT
ASSUME CS:CODE,DS:DATA,SS:STACK1
START:MOV AX,DATA
MOV DS,AX
MOV CH,0
MOV SI,OFFSET NUM1
MOV DI,OFFSET NUM2
MOV AH,NUM1L
MOV AL,NUM2L
CMP AH,AL
JAE CHA
XCHG SI,DI
XCHG AH,AL
CHA:SUB AH,AL
    MOV CL,AL
CLC
MOV BX,OFFSET OUTCOME1
LOP1:MOV AL,[DI]
PUSH CX
```

```
MOV CL,[SI]
    ADC CL,AL
MOV [BX],CL
INC BX
INC SI
INC DI
POP CX
LOOP LOP1
MOV CL,AH
JCXZ LOP3
LOP2:PUSH CX
MOV CL, BYTE PTR [SI]
ADC CL,0
MOV [BX],CL
POP CX
INC BX
    INC SI
    LOOP LOP2
LOP3:MOV CH,0
MOV SI,OFFSET NUM1
MOV DI,OFFSET NUM2
MOV AH,NUM1L
```

```
MOV AL,NUM2L
CMP AH,AL
JAE CHA1
XCHG SI,DI
XCHG AH,AL
CHA1:SUB AH,AL
      MOV CL,AL
CLC
MOV BX,OFFSET OUTCOME2
LOP11:MOV AL,[DI]
PUSH CX
MOV CL,[SI]
      SBB CL,AL
MOV [BX],CL
INC BX
INC SI
INC DI
POP CX
LOOP LOP11
MOV CL,AH
JCXZ LOP7
LOP21:PUSH CX
```

```
MOV CL, BYTE PTR [SI]
SBB CL,0
MOV [BX],CL
POP CX
INC BX
    INC SI
    LOOP LOP21
LOP7:MOV DI,OFFSET OUTCOME1
ADD DI,10H
MOV CX,11H
LOP4: MOV DL,[DI]
DEC DI
CMP N,0H
JNE LOP5
CMP DL,0H
JE LOP9
LOP5:PUSH DX
PUSH CX
MOV CL,4H
SHR DL,CL
POP CX
CMP DL,9H
```

```
JBE LOP6
ADD DL,7H
LOP6:ADD DL,30H
MOV AH,2
INT 21H
MOV N,1H
POP DX
AND DX,0FH
CMP DL,9H
JBE LOP8
ADD DL,7H
LOP8:ADD DL,30H
MOV AH,2
INT 21H
MOV DL,'H'
MOV AH,2
INT 21H
LOP9:LOOP LOP4
MOV N,0
MOV DI,OFFSET OUTCOME2
ADD DI,10H
MOV CX,11H
```

```
LOP14: MOV DL,[DI]
DEC DI
CMP N,0H
JNE LOP15
CMP DL,0H
JE LOP19
LOP15:PUSH DX
PUSH CX
MOV CL,4H
SHR DL,CL
POP CX
CMP DL,9H
JBE LOP16
ADD DL,7H
LOP16:ADD DL,30H
MOV AH,2
INT 21H
MOV N,1H
POP DX
AND DX,0FH
CMP DL,9H
JBE LOP18
```

```
ADD DL,7H
LOP18:ADD DL,30H
MOV AH,2
INT 21H
MOV DL,'H'
MOV AH,2
INT 21H
LOP19:LOOP LOP14
MOV AH,4CH
INT 21H
CODE ENDS
END START
```

## 实验 2

```
DATA SEGMENT
TISHI1 DB 'Please enter the first number:','$'
TISHI2 DB 'Please enter the second number:','$'
JIA DB '+','$'
JIAN DB '-','$'
CHENG DB '*','$'
CHU DB '/','$'
DENG DB '='','$'
YU DB '.....','$'
```

```
A DB 10
B DW 10
C DW 10000
F DW 0H
NUM1 DW 0H
NUM2 DW 0H
N DB 0H
N1 DB 4H
DATA ENDS
STACK1 SEGMENT STACK
DW 30H DUP(0)
STACK1 ENDS
CODES SEGMENT
ASSUME CS:CODES,DS:DATA
START:MOV AX,DATA
      MOV DS,AX
      MOV DX,OFFSET TISHI1
      MOV AH,09H
      INT 21H
LOOP2:MOV AX,NUM1
      MUL B
      MOV BX,AX
```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。

如要下载或阅读全文，请访问：

<https://d.book118.com/198052065123006106>