



**南昌理工學院**  
Nanchang Institute of Technology

**电子与信息学院**

# **微机原理与接口技术 实验指导书**

# 目 录

《微机原理与接口技术》课程实验报告（一） .....	1
《微机原理与接口技术》课程实验报告（二） .....	4
《微机原理与接口技术》课程实验报告（三） .....	6
《微机原理与接口技术》课程实验报告（四） .....	9

# 《微机原理与接口技术》课程实验报告（一）

## 【一、实验名称】

两个多位十进制数相加

## 【二、实验目的】

- 1、熟悉 DOS 系统命令；学习数据传送和算术运算指令的用法。
- 2、熟悉在 PC 机上建立、汇编、链接、调试和执行汇编语言程序的过程。
- 3、初步练习用 DEBUG 调试程序运行 8086 汇编语言程序的过程。

## 【三、实验内容和原理】

将两个 5 位十进制数相加。要求被加数和加数均以 ASCII 码形式各自顺序存放在以 DATA1 和 DATA2 为首地址的 5 个内存单元中（低位在前），结果送回 DATA1 处。另外，程序利用 DOS 的 9 号调用功能完成将被加数、加数和最后的和在显示器上的显示。

## 【四、实验条件】

微型计算机一台，软件 MASM

## 【五、实验过程】

1、用编辑软件,输入以下汇编语言源程序:

参考程序如下:

```
data segment
data1 db 32h, 38h, 30h, 36h, 37h, '$'
data2 db 33h, 35h, 33h, 38h, 32h, '$'
data ends

stac segment
sta db 100 dup (?)
stac ends

code segment
assume cs:code, ds:data, ss:stac, es:data
start proc far
push ds
xor ax, ax
push ax
mov ax, data
mov ds, ax
```

```

mov es, ax
mov ah, 9
mov dx, offset data1
int 21h
mov ah, 2
mov dl, '+'
int 21h
mov ah, 9
mov dx, offset data2
int 21h
mov si, offset data1
mov di, offset data2
mov bx, 5
push bx
push si
conv1:  and byte ptr[si+bx-1], 0fh
        and byte ptr[di+bx-1], 0fh
        dec bx
        jnz conv1
        mov cx, 5
        xor ax, ax
add1:   mov al, [si+4]
        mov bl, [di+4]
        adc al, bl
        aaa
        mov [si+4], al
        dec si
        dec di
        loop add1
        pop si
        pop bx
conv2:  or byte ptr[si+bx-1], 30h
        dec bx
        jnz conv2
        mov ah, 2
        mov dl, '='

```

```
int 21h
mov ah, 9
mov dx, si
int 21h
ret
start endp
code ends
end start
```

2.通过编译,连接形成可执行文件

3.运行。

## 【六、实验结果】

28067+35382=63449

## 【七、实验结果分析、讨论】

1、首先用 MASM 软件输入汇编语言源程序，然后对此源程序进行汇编，再用编译成目标文件，生成可执行文件，最后运行。如果程序运行结果不正确：

① 重复以上过程，并记录下修改经过，直至程序正确运行为止。

② 在步骤（1）中，如果程序运行结果不正确，可使用 DEBUG 程序进行动态调试，直到程序正确运行为止。

2、利用 DEBUG 程序修改被加数和加数，运行程序，并记录下程序运行结果。

将被加数修改为：31H，32H，33H，34H，35H

将加数修改为：35H，34H，33H，38H，39H

程序运行后，屏幕显示结果为：

12345+54389=66734

## 【八、实验结论】

利用 MASM 软件进行汇编，输入源程序后，需要不断地修改调试源程序，直到程序运行正确。

## 《微机原理与接口技术》课程实验报告（二）

### 【一、实验名称】

定时、计数器 8253A 应用

### 【二、实验目的】

学习 8253A 可编程定时、计数器与 8088CPU 的接口方法；了解 8253A 的工作方式；掌握 8253A 在各种方式下的编程方法。

### 【三、实验内容和原理】

#### 1、实验原理

8253A 的 A0、A1 接系统地址总线 A0、A1，故 8253A 有四个端口地址。8253A 的片选地址为 40H~4FH。因此，本实验仪中的 8253A 四个端口地址为 40H、41H、42H、43H，分别对应通道 0、通道 1、通道 2 和控制字。采用 8253A 通道 0，工作在方式 3，输入时钟 CLK0 为 1MHZ 的方波，并要求用接在 GATE0 引脚上的导线是接地或甩空来观察 GATE 对计数器的控制作用，用示波器观察输出波形。

#### 2、实验线路连接

(1) 8253A 的 GATE0 接+5V。

(2) 8253A 的 CLK0 插孔接分频器 74LS393 的 T4 插孔，分频器的频率源为 4MHZ。

### 【四、实验条件】

微型计算机一台，工具箱，示波器

### 【五、实验过程】

运行实验程序：单机时，实验程序起始地址为 F000：9180。在系统显示监控提示符“P.”时：输入 F000 按 F1 键输入 9180 按 EXEC 键。

将 CLK0 接 T6，OUT0 插孔连接一个发光的二极管。

附：实验参考程序：

```
ORG 08C0H      ;?  
L8253:  MOV DPTR,#0C003H  
        MOV A,#36H  
        MOVX @DPTR,A  
        MOV DPTR,#0C000H  
        MOV A,#00H  
        MOVX @DPTR,A  
        MOV A,#10H  
        MOVX @DPTR,A  
        SJMP $
```

END

## **【六、实验结果】**

当 CLK0 接 T6 时，发现发光的二极管的亮度是在改变的。当 CLK0 接 T7 时，发现发光的二极管的亮度的变化没有 CLK0 接 T6 时快。当 CLK0 接 T4 时，肉眼无法发现发光的二极管灯的亮度有改变。

## **【七、实验结果分析、讨论】**

发光二极管的亮度随着连接不同而发生改变。

## **【八、实验结论】**

不断地改变频率，发现发光的二极管的变化快慢与频率有关。

## 《微机原理与接口技术》课程实验报告（三）

### 【一、实验名称】

循环程序设计实验

### 【二、实验目的】

- 1.开始独立进行汇编语言程序设计；
- 2.掌握循环程序设计；
- 3.掌握最简单的 DOS 功能调用。

### 【三、实验内容和原理】

- 1.安排一个数据区,内存有若干个正数,负数和零.每类数的个数都不超过 9；
- 2.编写一个程序统计数据区中正数,负数和零的个数；
- 3.将统计结果在屏幕上显示。

### 【四、实验条件】

微型计算机一台，MASM 软件

### 【五、实验过程】

- 1.用编辑软件,输入以下汇编语言源程序:

源程序:

```
DAT SEGMENT
    Z   DW 0           ;数据中 0 的个数
    A   DW 0           ;数据中大于 0 的个数
    B   DW 0           ;数据中小于 0 的个数
    NUM DW 9,0,6,-4,-1,-5,5,-9,-8,-1,9      ;一组数据
    N   DW ($-NUM)/2   ;该组数据的个数
    TYPE0 DB 'The result is :', 0DH, 0AH, 'Number of zeros : $' ;显示一句话，展示 0 的个数
    TYPE1  DB 0DH, 0AH, 'Numbers above 0 : $'      ;显示一句话，展示大于 0 的个数
    TYPE2  DB 0DH, 0AH, 'Numbers below 0 : $'      ;显示一句话，展示小于 0 的个数
    EVEN
    LOCATEDW TYPE0, TYPE1, TYPE2
DAT ENDS
STA SEGMENT STACK
    DW 128 DUP(?)
STA ENDS
COD SEGMENT
```



```

    ASSUME    DS:DAT,SS:STA, CS:COD
STAR:
    MOV     AX, DAT
    MOV     DS, AX
    MOV     AX, STA
    MOV     SS, AX      ;初始化 DS、SS 指向 STA
    MOV     BX, OFFSET NUM
    MOV     CX, N
_START:MOV   AX, [BX]      ;取出 NUM 中的第一个数字
    CMP     AX, 0
    JZ     _Z              ;判断是 0，跳转到 Z 自加
    SAL    AX, 1
    JC     _B              ;判断是小于 0，跳转到 B 自加
    INC    WORD PTR A      ;否则，A 自加
    JMP    _OK
_B: INC    WORD PTR B
    JMP    _OK
_Z: INC    Z
_OK:
    ADD    BX, 2
    LOOP   _START         ;循环 CX 次判断
    MOV    CX, 3          ;显示 3 句话
    XOR    BX, BX
_SHOW:
    MOV    DX, LOCATE[BX] ;显示一句话
    MOV    AH, 09H
    INT    21H
    MOV    DX, Z[BX]
    ADD    DX, '0'
    MOV    AH, 02H
    INT    21H            ;加上 ASCII 码，显示 Z、A、B 的值
    ADD    BX, 2
    LOOP   _SHOW
    MOV    AX, 4C00H
    INT    21H            ;返回 DOS 系统
COD  ENDS
    END    STAR

```

2.通过编译,连接形成可执行文件。

## 【六、实验结果】

程序中数据为 NUM，是 9,0,6,-4,-1,-5,5,-9,-8,-1,9 一组数。

0 的个数: 1

正数的个数: 4

负数的个数: 6

## 【七、实验结果分析、讨论】

这个实验中利用循环统计正数、负数和零的个数。

## 【八、实验结论】

已经能够开始独立进行汇编语言程序设计了，通过上课和实验编程初步掌握了基本分支、循环程序的设计，数字的显示相关功能，即掌握了最简单的 DOS 功能调用。

但是用这种方法预见如果显示一串在程序中得到的数字，将会是比较繁琐的过程。

## 《微机原理与接口技术》课程实验报告（四）

### 【一、实验名称】

子程序设计实验

### 【二、实验目的】

- (1) 掌握分支程序及子程序设计的方法。
- (2) 进一步熟悉汇编语言源程序的上机过程。

### 【三、实验内容和原理】

有 10 个学生的成绩分别为：76、69、84、90、73、88、99、63、100、80，试编制一个程序统计 60 到 69、70 到 79、80 到 89、90 到 99 和 100 分的人数，分别放到五个存储单元中，然后在屏幕上显示。统计部分和显示部分要求用子程序实现。要求用堆栈进行参量传递。

### 【四、实验条件】

微型计算机一台，MASM 软件

### 【五、实验过程】

1.用编辑软件,输入以下汇编语言源程序:

源程序:

```
DATAS SEGMENT
```

```
SCORE db 76,69,84,90,73,88,99,63,100,80
```

```
STR1 db 'THE NUMBER OF 100 IS:','$'
```

```
STR2 db 'THE NUMBER OF 90-100 IS:','$'
```

```
STR3 db 'THE NUMBER OF 80-90 IS:','$'
```

```
STR4 db 'THE NUMBER OF 70-80 IS:','$'
```

```
STR5 db 'THE NUMBER OF 60-70 IS:','$'
```

```
COUNT EQU 10
```

```
GRADEA db ?
```

```
GRADEB db ?
```

```
GRADEC db ?
```

```
GRADEDE db ?
```

```
GRADEE db ?
```

DATAS ENDS

STACKS SEGMENT

STACKS ENDS

CODES SEGMENT

ASSUME CS:CODES,DS:DATAS,SS:STACKS

START:

MOV AX,DATAS

MOV DS,AX

LEA BX,SCORE

MOV CX,COUNT

A1:MOV AL,[BX]

CALL COMPARE

INC BX

LOOP A1

CALL OUTPUT

MOV AH,4CH

INT 21H

COMPARE PROC

PUSH BX

CMP AL,100

JE L1

CMP AL,90

JAE L2

CMP AL,80

JAE L3

CMP AL,70

JAE L4

CMP AL,60

JAE L5

L1:INC GRADEA

JMP DONE

L2:INC GRADEB

JMP DONE

L3:INC GRADEC

JMP DONE

L4:INC GRADED

JMP DONE

L5:INC GRADEE

JMP DONE

DONE: POP BX

RET

COMPARE ENDP

OUTPUT PROC

LEA DX,STR1

MOV AH,9

INT 21H

ADD GRADEA,30H

MOV DL,GRADEA

MOV AH,2

INT 21H

MOV DL,0AH

MOV AH,2

INT 21H

MOV DL,0DH

MOV AH,2

INT 21H

LEA DX,STR2

MOV AH,9

INT 21H

ADD GRADEB,30H

```
MOV DL,GRADEB
MOV AH,2
INT 21H
MOV DL,0AH
MOV AH,2
INT 21H
MOV DL,0DH
MOV AH,2
INT 21H
```

```
LEA DX,STR3
MOV AH,9
INT 21H
ADD GRADEC,30H
MOV DL,GRADEC
MOV AH,2
INT 21H
MOV DL,0AH
MOV AH,2
INT 21H
MOV DL,0DH
MOV AH,2
INT 21H
```

```
LEA DX,STR4
MOV AH,9
INT 21H
ADD GRADED,30H
MOV AH,2
MOV DL,GRADED
INT 21H
MOV DL,0AH
MOV AH,2
INT 21H
MOV DL,0DH
MOV AH,2
```

```
INT 21H

LEA DX,STR5
MOV AH,9
INT 21H
ADD GRADEE,30H
MOV DL,GRADEE
MOV AH,2
INT 21H
MOV DL,0AH
MOV AH,2
INT 21H

RET
OUTPUT ENDP
```

CODES ENDS

END START

2.通过编译,连接形成可执行文件。

## 【六、实验结果】

```
The number of 100      is : 1
The number of 90-100  is : 2
The number of 80-90   is : 3
The number of 70-80   is : 2
The number of 60-70   is : 2
```

## 【七、实验结果分析、讨论】

这个实验中采用了 DOS9 号、2 号功能调用，而且采用堆栈传递参数。

## 【八、实验结论】

在这个实验中，我进一步掌握了子程序的设计方法，还有基本的 DOS 9 号、2 号等功能调用。