

微型计算机原理 实验报告

班级： _____ XXXX _____

姓名： _____ XXXX _____

学号： _____ XXXXXXXXXXXX _____

实验二 分支程序设计与排序

一、实验目的

1. 熟悉 8086 汇编语言程序的基本格式，以及汇编、链接的基本过程。
2. 熟悉汇编分支程序与循环程序的编写。
3. 掌握使用各种寻址指令，和运算类指令编程及调试方法。

二、实验内容

1. 有一个 10 字节的数组，其值分别是 06H, F2H, 5AH, F4H, 97H, 64H, BBH, 7FH, 0FH, D8H。编程并显示结果：
 - (1) 如果数组是无符号数，按大小排序从内存单元(4000H)开始连续存入排序后数组，并求出最大值，并显示；
 - (2) 如果数组是有符号数，按大小排序从内存单元(4000H)开始连续存入排序后数组，并求出最大值，并显示。

三、实验程序及设计流程图

1. 无符号数组排序

(1) 源程序

```
DATA SEGMENT
    STRING DB 06H,F2H,5AH,F4H,97H,64H,BBH,7FH,0FH,D8H
    NUM $-STRING
    RESULT DB DUP(?)
DATA ENDS

CODE SEGMENT
    ASSUME CS:CODE,DS:DATA
START:MOV AX,DATA
    MOV DS,AX
    MOV CX,NUM
    DEC CX
OUID:MOV DX,CX
    MOV SI,0
INSID:MOV AL,STRING[SI]
    CMP AL,STRING[SI+1]
    JAE LAB1
    XCHG AL,STRING[SI+1]
    MOV STRING[SI],AL
```

```

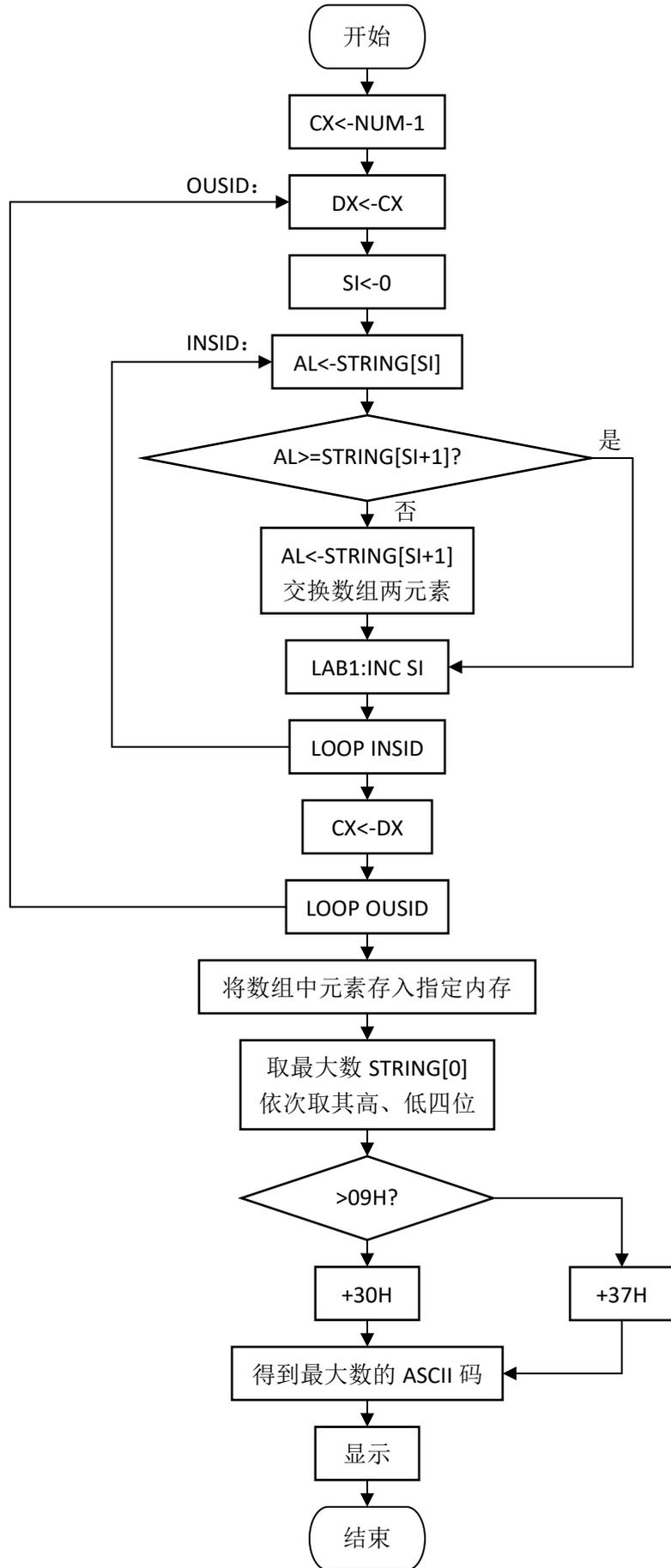
LAB1:INC SI
      LOOP INSID
      MOV CX,DX
      LOOP OUSID
      MOV CX,NUM-STRING
      MOV DI,0
      MOV SI,OFFSET STRING
AGAIN:LODSB
      MOV [4000H+DI],AL
      INC DI
      LOOP AGAIN
      MOV AL,STRING[0]
      AND AL,0F0H
      MOV CL,4
      SHR AL,CL
      CMP AL,09H
      JBE LAB2
      ADD AH,37H
      JMP LAB3
LAB2:ADD AH,30H
LAB3:MOV RESULT[0],AL
      MOV AL,STRING[0]
      AND AL,0FH
      CMP AL,09H
      JBE LAB4
      ADD AL,37H
      JMP LAB5
LAB4:ADD AL,30H
LAB5:MOV RESULT[1],AL
      LEA DX,RESULT
      MOV AH,09H
      INT 21H
      MOV AH,4CH
      INT 21H
CODE ENDS
END START

```

(2) 程序流程图

本实验中采用冒泡排序法进行由大到小的排序，其中需要N-1次外层循环，每次外层循环中嵌套N-1次内层循环。内、外两层循环均由LOOP跳转指令实现，而控制循环次数的只有CX一个寄存器，因此要注意在内、外两层循环切换时进行参数的传递。

当排序完成后，需要取出最大数进行显示，此时需将其转换为ASCII码，并调用DOS功能实现在屏幕上显示。程序流程图如下：



2. 有符号数组排序

(1) 源程序

有符号数组排序的程序与无符号数组排序的程序相似，只是需要把有条件跳转的判断条件改为与有符号数对应的条件，源程序如下：

```
DATA SEGMENT
    STRING DB 06H,F2H,5AH,F4H,97H,64H,BBH,7FH,0FH,D8H
    NUM $-STRING
    RESULT DB DUP(?)
DATA ENDS

CODE SEGMENT
    ASSUME CS:CODE,DS:DATA
START:MOV AX,DATA
    MOV DS,AX
    MOV CX,NUM
    DEC CX
OUSID:MOV DX,CX
    MOV SI,0
INSID:MOV AL,STRING[SI]
    CMP AL,STRING[SI+1]
    JGE LAB1
    XCHG AL,STRING[SI+1]
    MOV STRING[SI],AL
LAB1:INC SI
    LOOP INSID
    MOV CX,DX
    LOOP OUSID
    MOV CX,NUM-STRING
    MOV DI,0
    MOV SI,OFFSET STRING
AGAIN:LDSB
    MOV [4000H+DI],AL
    INC DI
    LOOP AGAIN
    MOV AL,STRING[0]
    AND AL,0F0H
    MOV CL,4
    SHR AL,CL
    CMP AL,09H
    JBE LAB2
    ADD AL,37H
    JMP LAB3
LAB2:ADD AL,30H
LAB3:MOV RESULT[0],AL
```

```

MOV AL,STRING[0]
AND AL,0FH
CMP AL,09H
JBE LAB4
ADD AL,37H
JMP LAB5
LAB4:ADD AL,30H
LAB5:MOV RESULT[1],AL
LEA DX,RESULT
MOV AH,09H
INT 21H
MOV AH,4CH
INT 21H
CODE ENDS
END START

```

(2) 流程图

与无符号数组排序相同。

四、实验数据及结果分析

1. 无符号数组排序

排序前数组 `STRING` 存储在以 `[0000H]` 为起始的内存单元里，其初始状态如下

```

ds:0000 06 F2 5A F4 97 64 BB 7F
ds:0008 0F D8 0A 00 00 00 00 00

```

排序后的数组重新存储在以 `[4000H]` 为起始的内存单元里，排序后结果如下，最大值为 `F4H`。其中 `ds:000AH` 中存储的为数组元素个数，`ds:000CH` 和 `ds:000DH` 中存储的为要显示的最大数的 ASCII 码，即 `4634H`。

```

ds:0000 F4 F2 D8 BB 97 7F 64 5A
ds:0008 0F 06 0A 00 46 34 00 00

ds:4000 F4 F2 D8 BB 97 7F 64 5A
ds:4008 0F 06 00 00 00 00 00 00

```

2. 有符号

排序前数组 `STRING` 存储在以 `[0000H]` 为起始的内存单元里，其初始状态如下

```

ds:0000 06 F2 5A F4 97 64 BB 7F
ds:0008 0F D8 0A 00 00 00 00 00

```

排序后的数组重新存储在以 `[4000H]` 为起始的内存单元里，排序后结果如下，最大值为 `7FH`。其中 `ds:000AH` 中存储的为数组元素个数，`ds:000CH` 和 `ds:000DH` 中存储的为要显示的最大数的 ASCII 码，即 `3746H`。

```

ds:0000 7F 64 5A 0F 06 F4 F2 D8
ds:0008 BB 97 0A 00 37 46 00 00

ds:4000 7F 64 5A 0F 06 F4 F2 D8
ds:4008 BB 97 00 00 00 00 00 00

```

