# 

**计算机科学与技术学院**

**2016-2017学年第2学期**

**《汇编语言程序设计》**

**实验8：汇编综合成绩设计**

**专业： 信息安全**

**学号： 150341221**

**姓名： 路伟饶**

**教师： 王家亮**

**完成日期：2017年6月25日**

1. 实验要求
2. 程序说明。说明程序的功能、结构。
3. 调试说明。包括上机调试的情况、上机调试步骤、调试所遇到的问题是如何解决的，并对调试过程中的问题进行分析，对执行结果进行分析。
4. 写出源程序清单和执行结果，并给出程序流程图。
5. 分析实验结果及所遇到的问题的解决方法。
6. 体会和建议。
7. 实验目的
8. 通过前面所学的汇编语言结构(顺序、分支、循环和子程序结构)以及 DOS 功能调用等基本的程序结构来设计一个综合程序，达到学以致用，举一反三;
9. 掌握模块化程序的设计方法;
10. 掌握综合程序的编制及调试方法。
11. 程序说明
12. 编程实现由键盘输入任意一个字符，将该字符的 ASCII 码值显示在屏幕上(要求:分别以二进制和十六进制形式显示)。

DATA SEGMENT

NUMBERS DB '0','1','2','3','4','5','6','7','8','9','A','B','C','D','E','F'

MESSAGE DB 'ASCII Shower',0AH,'$'

EXITMESSAGE DB 'BYE~ $'

DATA ENDS

STACK SEGMENT STACK

STACK ENDS

CODE SEGMENT

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

;子程序功能：输出一位十六进制数字

;入口参数：DL=数字

PrintHexNumber PROC

PUSH BX

PUSH SI

PUSH AX ;保存寄存器值

AND DX,000FH ;取最低四位

LEA BX,NUMBERS ;装载数字

MOV SI,DX; ;将数字设置为偏移量

MOV DL,[BX+SI] ;根据偏移量可以在数字中输出对应的

MOV AH,2

INT 21H ;输出数字

POP AX

POP SI

POP BX ;还原寄存器值

RET ;函数返回

PrintHexNumber ENDP

;子程序功能：输出一位二进制数字

;入口参数：DL=数字

PrintBinNumber PROC

PUSH BX

PUSH SI

PUSH AX ;保存寄存器值

AND DX,0001H ;取最低一位

LEA BX,NUMBERS ;装载数字

MOV SI,DX; ;将数字设置为偏移量

MOV DL,[BX+SI] ;根据偏移量可以在数字中输出对应的

MOV AH,2

INT 21H ;输出数字

POP AX

POP SI

POP BX ;还原寄存器值

RET ;函数返回

PrintBinNumber ENDP

START:

MOV AX,DATA ;设置数据段

MOV DS,AX

LEA DX,MESSAGE

MOV AH,9

INT 21H

for:

MOV CX,1 ;循环内设置CX循环次数，循环不停止

MOV AH,1

INT 21H ;输入一个字母，对应ASCII码值存入AL

PUSH DX

PUSH AX

MOV AH,2

MOV DL,0AH

INT 21H

POP AX

POP DX

PUSH CX ;保存CX寄存器

MOV CX,8

ROL AX,CL ;循环左移8位，将低8位的内容移入高8位

POP CX

MOV CX,2 ;十六进制，需要取两位数

s16:

PUSH CX

MOV CX,4

ROL AX,CL ;循环左移4位，将最高位数字送入最低位

POP CX

MOV DX,AX ;数字转移

CALL PrintHexNumber ;调用子程序输出

LOOP s16

PUSH DX

PUSH AX

MOV DL,0AH

MOV AH,2

INT 21H

POP AX

POP DX

MOV CX,8

ROL AX,CL ;循环左移8位，准备输出二进制

s2:

ROL AX,1

MOV DX,AX

CALL PrintBinNumber ;二进制输出

LOOP s2

PUSH DX

PUSH AX

MOV DL,0AH

MOV AH,2

INT 21H

POP AX

POP DX

CMP AL,'Q' ;输入字母Q或者q来退出

JZ exit

CMP AL,'q'

JZ exit

LOOP for

exit:

LEA DX,EXITMESSAGE

MOV AH,9

INT 21H

MOV AH,4CH

INT 21H

CODE ENDS

END START

1. 编写一个程序,实现把存放在 BX 寄存器内的二进制数以十六进制数的形式显示在屏幕上。

DATA SEGMENT

NUMBERS DB '0','1','2','3','4','5','6','7','8','9','A','B','C','D','E','F'

DATA ENDS

STACK SEGMENT STACK

STACK ENDS

CODE SEGMENT

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

;子程序功能：输出一位十六进制数字

;入口参数：DL=数字

PrintHexNumber PROC

PUSH BX

PUSH SI

PUSH AX ;保存寄存器值

AND DX,000FH ;取最低四位

LEA BX,NUMBERS ;装载数字

MOV SI,DX; ;将数字设置为偏移量

MOV DL,[BX+SI] ;根据偏移量可以在数字中输出对应的

MOV AH,2

INT 21H ;输出数字

POP AX

POP SI

POP BX ;还原寄存器值

RET ;函数返回

PrintHexNumber ENDP

START:

MOV AX,DATA

MOV DS,AX

MOV BX,1234H

MOV CX,4

for:

PUSH CX

MOV CL,4

ROL BX,CL

POP CX

MOV DX,BX

CALL PrintHexNumber

LOOP for

MOV AH,4CH

INT 21H

CODE ENDS

END START

1. 现在 block 数据区有 20 个字数据，按要求编写程程序: (1)将正数和负数分开存放; (2)统计其中的正数和负数个数，并将结果放在各区前面。

DATA SEGMENT

BLOCK DW 12,-5,7,90,123,-12,36,314,129,10,565,234,2,3,-24,124,112,-125,23,0

POSNUM DW 21 DUP(0)

NEGNUM DW 21 DUP(0)

DATA ENDS

STACK SEGMENT STACK

STACK ENDS

CODE SEGMENT

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

START:

MOV AX,DATA

MOV DS,AX

LEA SI,BLOCK

CLD

MOV CX,20

MOV BX,0

MOV BP,0

for:

LODSW

CMP AX,0

JS negn

JMP posn

posn:

ADD BX,2

MOV [POSNUM+BX],AX

JMP continue

negn:

ADD BP,2

MOV [NEGNUM+BP],AX

JMP continue

continue:

LOOP for

exit:

SAR BX,1

SAR BP,1

MOV POSNUM,BX

MOV NEGNUM,BP

MOV AH,4CH

INT 21H

CODE ENDS

END START

1. 在以 BUF 为首址的字存储区中存放有 N 个有符号数，现将它们按由小到大的顺序排列后再放 回 BUF 存储区中，试编程实现。

DATA SEGMENT

BUF DB 10,39,18,21,79,65,33,43,22,62

COUNT DW $-ARRAY

DATA ENDS

STACK SEGMENT STACK

STACK ENDS

CODE SEGMENT

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

START:

MOV AX,DATA

MOV DS,AX

MOV AX,COUNT

SUB AX,1

MOV CX,AX

MOV SI,1

i:

MOV AL,[BUF+SI]

MOV DX,CX

MOV CX,SI

k:

MOV DI,CX

MOV AH,[BUF+DI-1]

SUB AH,AL

JS bre

MOV BL,[BUF+DI-1]

MOV [BUF+DI],BL

JMP \_in

bre:

JMP \_ou

\_in:

LOOP k

\_ou:

MOV BL,AL

MOV [BUF+DI],BL

MOV CX,DX

INC SI

LOOP i

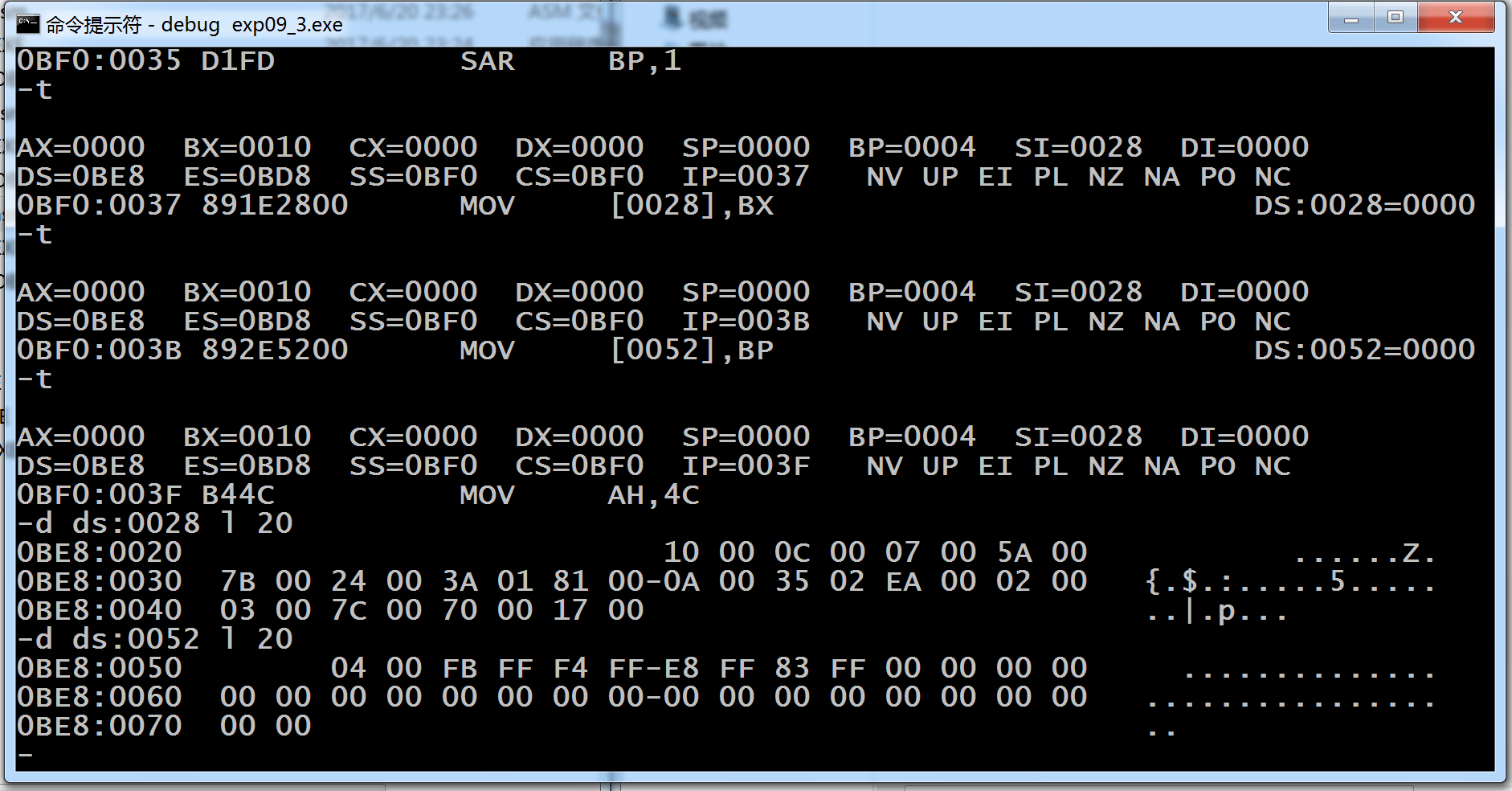
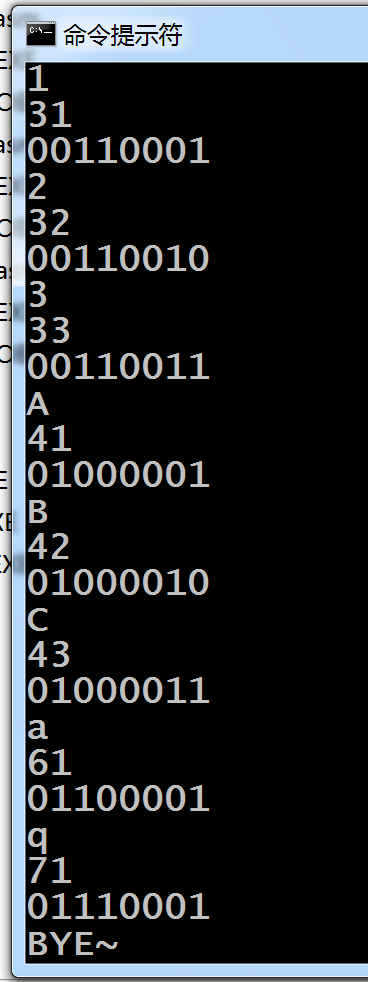
MOV AH,4CH

INT 21H

CODE ENDS

END START

1. 上机调试说明
2. 本次上机使用的是Windows 7 32位操作系统，在调试过程中，第一个程序的调试比较困难，主要问题是忽略了INT 21H指令中会改变寄存器的值，所以使用栈来保存寄存器的值是非常重要的。
3. 在汇编语言编程过程中，编写恰当的子程序有利于程序的编写和阅读。
4. 在使用汇编进行中断调用，输入输出内容的时候，应该注意汇编语言的特殊性，其对控制台的操作比较简单，如果编写的程序不合理，很容易造成显示不正常的情况。
5. 编写程序一定要设置退出的方法，不然在DOS窗口下无法退出程序，非常麻烦。
6. 写出源程序清单及执行结果



1. 心得体会

本次实验综合应用汇编程序设计的知识，特别是循环、判断、串操作、子程序、中断调用这些核心知识点。通过本次实验，对本学期学的知识起到了一个复习总结作用。也为未来继续深入学习其他种类的汇编语言打下良好的基础。