



华南理工大学

SOUTH CHINA UNIVERSITY OF TECHNOLOGY

# 微机原理实验报告

实验三

-----

评阅分数：\_\_\_\_\_

## 一、实验目的

- 1、熟悉 80X86 基本指令集。
- 2、掌握汇编语言源程序的编写方法：程序结构、返回 DOS 的方法、段地址的初始化、存储单元的访问等。
- 3、掌握常用的代码转换编程。
- 4、掌握汇编程序的 DEBUG 动态调试。

## 二、实验题目

题目 1、码制转换：将字节数据转换为 ASCII 码表示的二进制数，并显示输出。字节数据的值域为 0~255，可转换为八位二进制数。例如：字节数据为 79h，则转换为：‘01111001’

①已知数据段的定义：

```
DATA    SEGMENT
        org 2000h
        num db 79H
        org 2010h
        res db 8 dup(?), '$'
DATA    ENDS
```

②debug 调试：

- 1) 查看运行结果应该为：在 2010h~2017h 存储单元为：30, 31, 31, 31, 31, 30, 30, 31。
- 2) 修改 num，运行看结果。

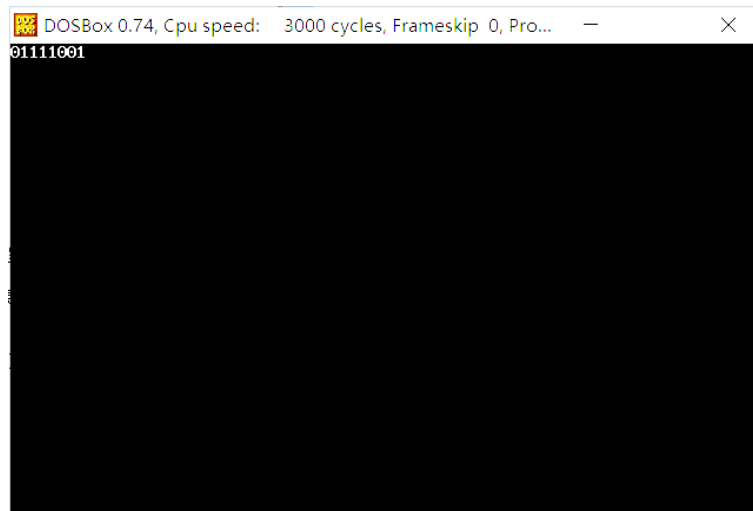
源代码:

```
DATA    SEGMENT
        org 2000h
        num db 79H
        org 2010h
        res db 8 dup(?),'$'
DATA    ENDS

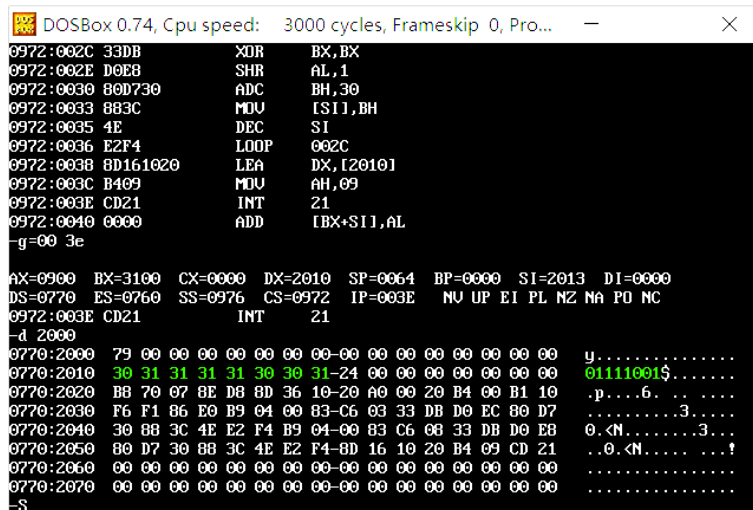
STACK   SEGMENT PARA STACK 'STACK'
        DB 100 DUP (?)
STACK   ENDS

CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA,SS:STACK
START:  MOV AX,DATA
        MOV DS,AX
;初始赋值
        LEA SI,res
        MOV AL,[num]
        MOV AH,0H
        MOV CL,10H
        DIV CL           ; 将79H 分成7H 和9H
        XCHG AH,AL       ; 交换 AL 和 AH, 使得 AX 为7H 和9H
; 处理 AH
        MOV CX,4H
        ADD SI,3
HEIGHT: XOR BX,BX
        SHR AH,1         ; 右移
        ADC BH,30H       ; 将结果修改为 ASCII 码
        MOV [SI],BH      ; 保存结果
        DEC SI
        LOOP HEIGHT
; 处理 AL
        MOV CX,4H
        ADD SI,8
LOWER:  XOR BX,BX
        SHR AL,1         ; 右移
        ADC BH,30H       ; 将结果修改为 ASCII 码
        MOV [SI],BH      ; 保存结果
        DEC SI
        LOOP LOWER
; 在屏幕上显示结果
        LEA DX,res
        MOV AH,09H
        INT 21H
CODE    ENDS
        END START
```

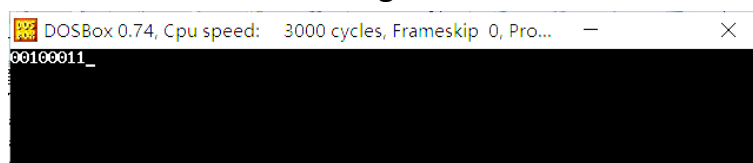
## Debug 结果:



## 运行结果



## Debug 结果



修改为 23H 之后输出的结果，理论上应该要输出 00100011

如上图，对于原始定义，能够正确输出结果，同时，在将 num 修改为 23H 后也能输出正确的结果。程序运行无误。

题目 2、码制转换：将字数据转换为 ASCII 码表示的十进制数，并显示输出。字数据的值域为 0~65535，最多可转换为五位十进制数。例如：字数据为 0ffffh，则转换为：‘65535’

①已知数据段的定义：

```
DATA    SEGMENT
    org  2000h
    num  dw  65535
    org  2010h
    res  db  5 dup(?), '$'
DATA    ENDS
```

②debug 调试：

- 1) 查看运行结果应该为：在 2010h~2014h 存储单元为：36, 35, 35, 33, 35。
- 2) 修改 num，运行看结果。

源代码：

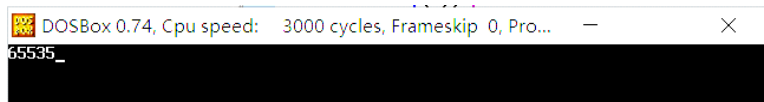
```
DATA    SEGMENT
    org  2000h
    num  dw  65535
    org  2010h
    res  db  5 dup(?), '$'
DATA    ENDS

STACK   SEGMENT PARA STACK 'STACK'
    DB  100 DUP (?)
STACK   ENDS

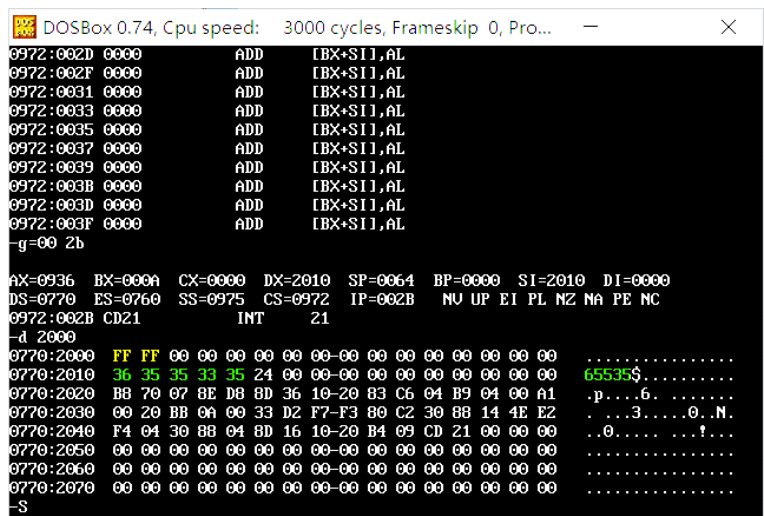
CODE    SEGMENT
    ASSUME CS:CODE,DS:DATA,SS:STACK
START:  MOV AX,DATA
        MOV DS,AX
; 初始赋值
        LEA SI,res
        ADD SI,4H
        MOV CX,4H
        MOV AX,[num]
        MOV BX,0AH
```

```
NEXT:  XOR DX,DX      ; 将 DX 和 CF 清零
      DIV BX         ; 除10取余
      ADD DL,30H     ; 将结果修改为对应的 ASCII 码
      MOV [SI],DL    ; 把结果保存到 res
      DEC SI
      LOOP NEXT
; 处理剩余一位，也就是 AL
      ADD AL,30H
      MOV [SI],AL
      LEA DX,res
      MOV AH,09H     ; 输出结果到屏幕
      INT 21H
CODE  ENDS
      END START
```

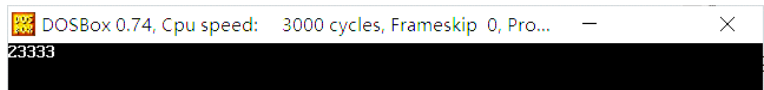
Debug 结果:



直接运行结果



debug 结果



修改数据为 23333 后输出的结果

根据上面的图片，可以看出能够正确地输出需要的结果，满足题目的要求。

3、码制转换：将字数据转换为 ASCII 码表示的十六进制数，并显示输出。字数据的值域为 0~65535，最多可转换为四位十六进制数。例如：字数据为 0ffffh，则转换为：‘FFFF’

①已知数据段的定义：

```
DATA    SEGMENT
    org  2000h
    num  dw  54A8h
    org  2010h
    res  db  4 dup(?),'$'
DATA    ENDS
```

②debug 调试：

1) 查看运行结果应该为：在 2010h~2013h 存储单元为：35，34，41，38。

2) 修改 num，运行看结果。

源代码：

```
DATA    SEGMENT
    org  2000h
    num  dw  54A8h
    org  2010h
    res  db  4 dup(?),'$'
DATA    ENDS

STACK   SEGMENT PARA STACK 'STACK'
    DB  100 DUP (?)
STACK   ENDS

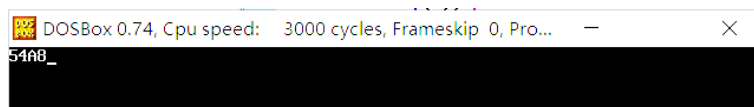
CODE    SEGMENT
    ASSUME CS:CODE,DS:DATA,SS:STACK
START:  MOV AX,DATA
        MOV DS,AX
; 初始赋值
        LEA SI,res
        ADD SI,3H
        MOV AX,[num]
        MOV BX,10H
        MOV CX,3H
```

```

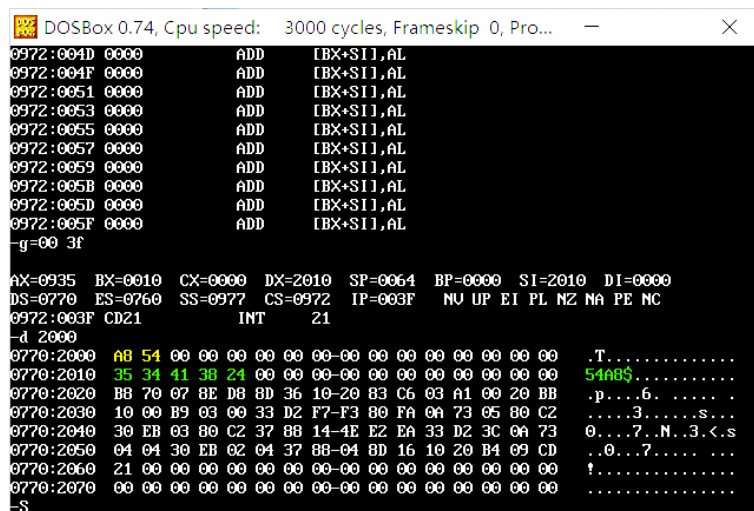
NEXT:  XOR DX,DX      ; 将 dx 和 cf 清零
        DIV BX        ; 除16取余数
        CMP DL,0AH    ; 判断是数字还是字母
        JAE LETTER
        ADD DL,30H     ; 数字加30H
        JMP SAVE
LETTER: ADD DL,37H     ; 字母加37H
SAVE:   MOV [SI],DL    ; 将修改后的 ASCII 码存到 res
        DEC SI
        LOOP NEXT
; 处理剩余的一位，也就是 AL 对应的内容
        XOR DX,DX
        CMP AL,0AH
        JAE LETTERA
        ADD AL,30H
        JMP SAVEA
LETTERA:ADD AL,37H
SAVEA:  MOV [SI],AL
        LEA DX,res
        MOV AH,09H    ; 输出结果
        INT 21H
CODE    ENDS
        END START

```

Debug 结果:

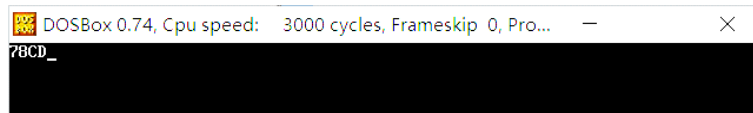


能够正确输出内容



debug 结果





修改 num 为 78CD 后的输出结果

### 三、实验总结

通过这次试验，让我更进一步的记住 **80x86** 的指令集的使用。对于码制转换也更加熟悉。第二、三题中运用了“除 **r** 取余法”来求对应的结果，这个方法在平时中可能很少用到，但在现在显得很实用。