**汇编语言与逆向技术实验报告**

**Lab2-dec2hex**

**学号：2112060 姓名：孙蕗 专业：信息安全**

1. **实验目的**
   1. 熟悉汇编语言的数据传送、寻址和算术运算；
   2. 熟悉汇编语言过程的定义和使用；
   3. 熟悉十进制和十六进制的数制转换
2. **实验环境**

MASM32编译环境

Windows命令行窗口

1. **实验内容**

编写汇编程序dec2hex.asm，编译成dec2hex.exe。dec2hex.exe的功能是将Windows命令行输入的十进制无符号整数，转换成对应的十六进制整数，输出在Windows命令行中，如图1所示。

输入的十进制无符号整数的范围是**0到4294967295（232-1）。**

输出对应的十六进制整数，对应的范围是**00000000h到FFFFFFFFh。**

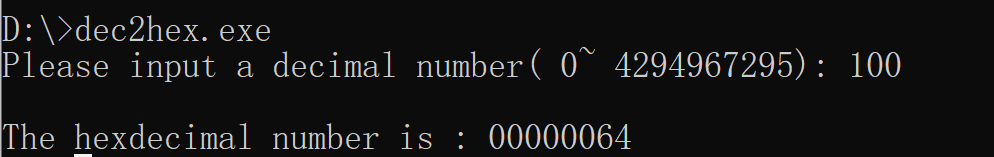


图1. dec2hex.exe将十进制100转换成十六进制00000064

3.1 使用StdIn函数获得用户输入的十进制整数。StdIn函数的定义在\masm32\include\masm32.inc，库文件是\masm32\lib\masm32.lib。StdIn函数的定义“StdIn PROTO :DWORD,:DWORD”，有两个参数，第一个是内存存储空间的起始地址，第二个是内存存储空间的大小。函数的例子：

.data

buf BYTE 20 DUP(0)

.code

invoke StdIn, addr buf, 20

invoke StdOut, addr buf

3.2 用户输入的十进制数对应的ASCII编码字符串存储在内存中，编写过程dec2dw，将ASCII字符串转换成DWORD数据。例如，将字符串“100”转换成DWORD数据00000064h。

3.3 编写过程dw2hex，将DWORD数据转换成十六进制数的ASCII字符串。例如，将DWORD数据00000064h转换成ASCII字符串“00000064”

3.4 使用StdOut函数在Windows命令函中输出十六进制整数的ASCII字符串。StdOut函数的定义在\masm32\include\masm32.inc，库文件是\masm32\lib\masm32.lib。StdOut函数的定义“StdOut PROTO :DWORD”，只有一个参数，是内存存储空间的起始地址。函数使用的例子同StdIn函数的例子。

3.5 使用ml将dec2hex.asm文件汇编到dec2hex.obj目标文件，编译命令：“\masm32\bin\ml /c /coff dec2hex.asm”

3.6 使用link将目标文件dec2hex.obj链接成dec2hex.exe可执行文件，链接命令：“\masm32\bin\link /SUBSYSTEM: CONSOLE dec2hex.obj”

1. **源代码**

.386

.MODEL flat, stdcall

OPTION casemap :none

INCLUDE \masm32\include\windows.inc

INCLUDE \masm32\include\kernel32.inc

INCLUDE \masm32\include\masm32.inc

INCLUDELIB \masm32\lib\kernel32.lib

INCLUDELIB \masm32\lib\masm32.lib

.data

str1 BYTE "Please input a decimal number(0~4294967295):", 0 ;输入提示

str2 BYTE "The hexdecimal number is:", 0 ;输出提示

mmc BYTE "0123456789ABCDEF", 0

dec\_num BYTE 10 DUP(0), 0 ;存十进制字符串

dw\_num DWORD 0 ;十进制数字

dec\_cnt DWORD 0

hex\_cnt DWORD 0

v16 BYTE 10h ;数16

v10 DWORD 0Ah ;数10

power DWORD 1

one BYTE 0, 0

oneAH BYTE 0

oneECX DWORD 0

oneESI PDWORD 0

.code

main PROC

INVOKE StdOut, ADDR str1 ;输出输入提示字符串

INVOKE StdIn, ADDR dec\_num, 10 ;输入十进制字符串

call dec2dw; ;调过程dec2dw

INVOKE StdOut, ADDR str2; ;输出输出提示字符串

call Dw2hex; ;调过程dw2hex

INVOKE ExitProcess, 0 ;结束

main ENDP

dec2dw PROC

mov esi, OFFSET dec\_num; ;把变量dec\_num的偏移地址放到寄存器esi中

L1:

inc dec\_cnt ;+1

inc esi ;+1

mov eax, [esi]

cmp al, 0 ;看有没有到字符串尾

je L2 ;有就跳出

jmp L1

L2:

mov ecx, dec\_cnt ;记录几个字符数字

L3:

sub dec\_num[ecx-1], '0' ;处理成数字

mov eax, 0 ;清零

mov al, BYTE ptr dec\_num[ECX-1] ;从头开始

mul power

ADD dw\_num, eax

mov eax, power

mul v10 ;移位乘十加个位

mov power, eax

loop L3

RET

dec2dw ENDP

Dw2hex PROC

mov esi, OFFSET dw\_num+3

mov ecx, 4

L4:

mov ax, 0

mov al, BYTE ptr [ESI]

div v16 ;除16

mov oneAH, ah

mov oneECX, ecx ;复制赋值

xchg esi, oneESI ;交换内容

;0-9转字符

mov esi, OFFSET mmc

mov one, al

movzx ebx, one ;movzx将源操作数取出来置于目的操作数,目的操作数其余位用0填充

add esi, ebx

mov bl, BYTE ptr [ESI]

mov one, bl

INVOKE StdOut, ADDR one

;10-15转字符

mov esi, OFFSET mmc

mov ah, oneAH

mov one, ah

movzx ebx, one

add esi, ebx

mov bl, BYTE ptr [ESI]

mov one, bl ;字符

INVOKE StdOut, ADDR one

xchg oneESI, esi ;交换内容

mov ecx, oneECX ;个数减少

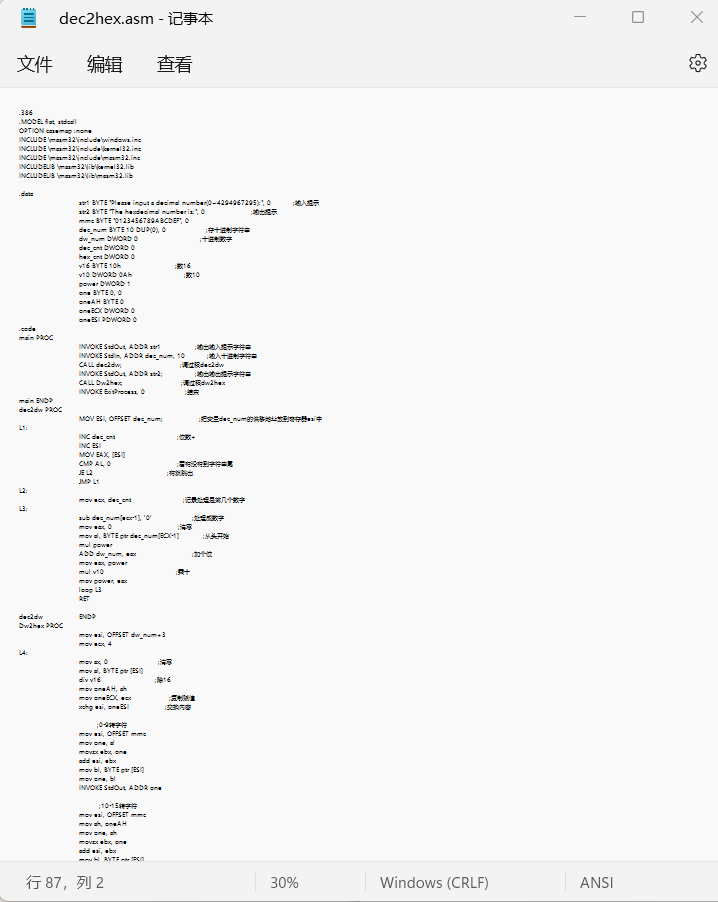
dec esi

LOOP L4

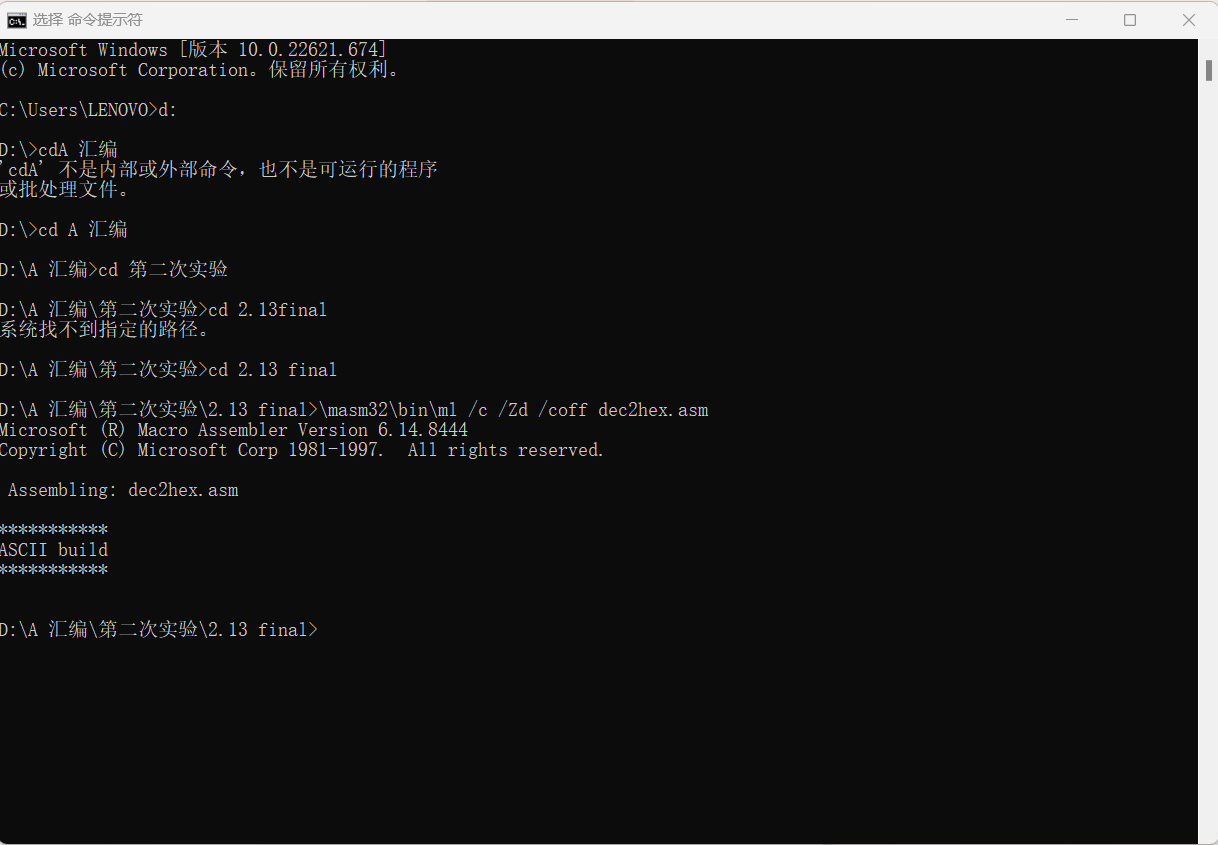
RET

Dw2hex ENDP

END main

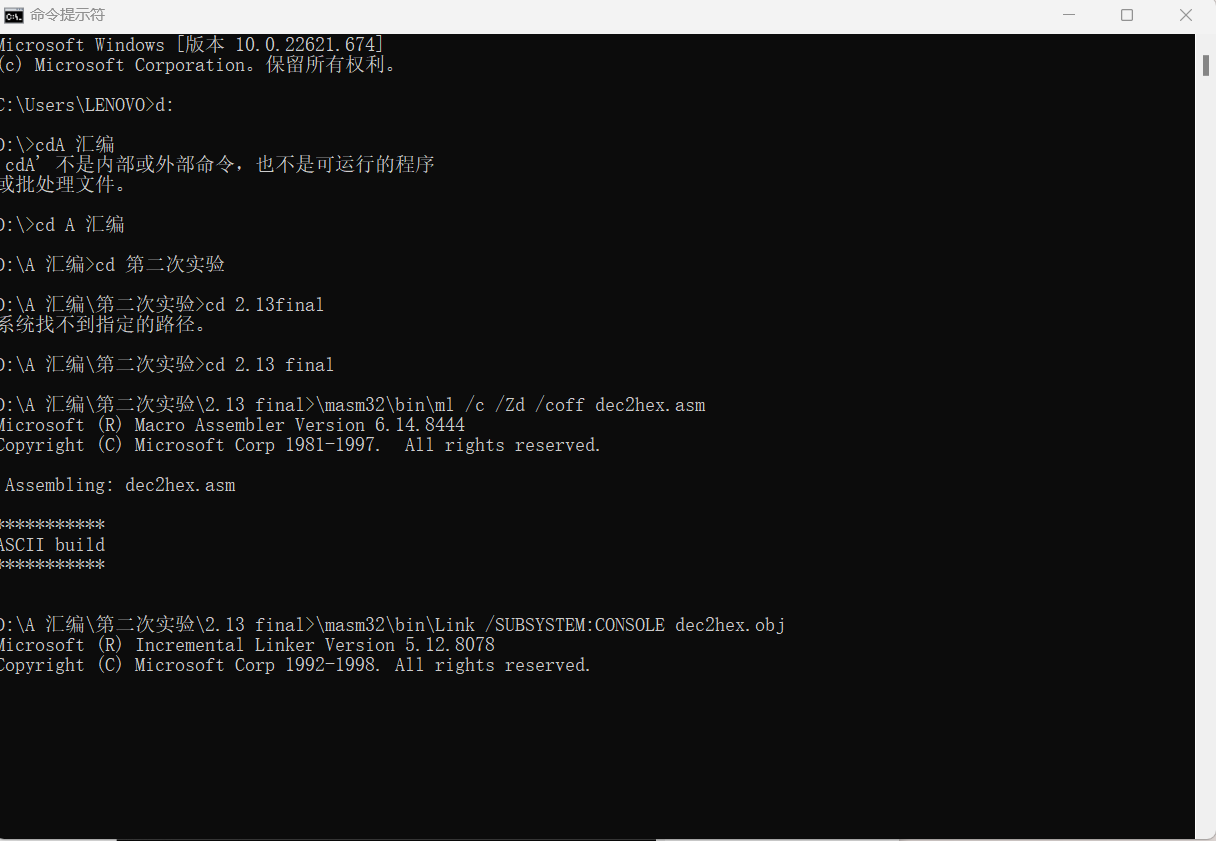
1. **实验步骤**
2. 源文件：用文本编辑器编写的asm文本文件
3. ****汇编：用汇编程序（\masm32\bin\ml.exe）对源程序进行汇编，形成目标文件（.obj），格式如下：

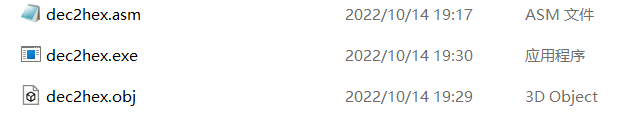
“\masm32\bin\ml /c /Zd /coff dec2hex.asm”



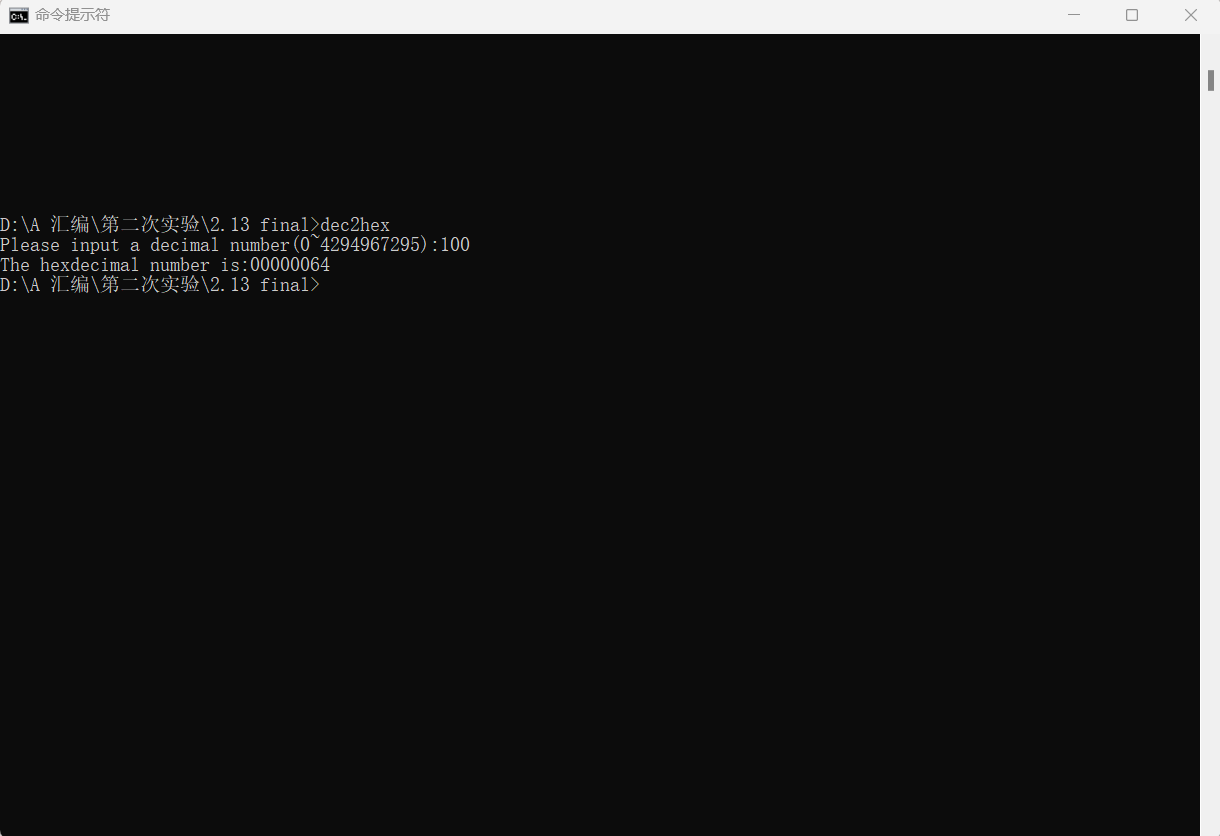
1. 连接：用连接程序（\masm32\bin\link.exe）对目标程序进行连接，形成可执行文件（.exe），格式如下

“\masm32\bin\Link /SUBSYSTEM:CONSOLE hello\_console.obj”



****

1. **测试说明**

****