

天津商业大学信息工程学院专业实验室学生实验报告

实验日期：2013 年 12 月 20 日 实验时间：10: 05—11: 40
课程名称：汇编语言程序设计 任课教师：苗序娟 实验成绩：
专业班级：软件工程 12-01 班 学生姓名：王靖伟
实验项目名称：子程序设计

实验内容及结果：

实验要求（教师给出）：

通过子程序设计，掌握汇编程序设计思路和编写方法，并通过实验观察和分析程序的执行结果。熟练子程序设计方法和子程序链接。

- 1、写出设计思路和程序源码
- 2、实验内容用截图形式记录实验结果
- 3、写出实验结果分析

实验过程、结果（实验过程、图、代码、结果及实验收获等）：

一、实验过程

1.题目：

设计程序。用菜单形式实现十到二进制、十到八进制、十到十六进制的转换。采用子程序调用等模块化程序设计方法。

2.设计思路：

- 1) 用 DOS 中断调用的 1 号功能输入数据，用 2 号功能显示结果，9 号功能显示提示信息。
- 2) 根据选择执行某程序段。
- 3) 将要显示的数值变为 ASCII 码。
- 4) 设计一个主程序 main，四个子程序 prog1、prog2、prog3、prog4，用五个模块分别保存，实现远程的访问与调用。
- 5) 主程序的功能是调用四个子程序。
- 6) prog1 的功能是 65535 以内十进制转换二进制。
- 7) prog2 的功能是 65535 以内十进制转换八进制。
- 8) prog3 的功能是 255 以内十进制转换十六进制。
- 9) prog4 的功能是判断并退出程序。

3.程序代码:

表 1. 5-1.asm 代码

行号	5-1.asm 代码
1	extrn prog1:far,prog2:far,prog3:far,prog4:far
2	public infor,mess0,mess1,mess2,mess3,mess4,mess5,x,x1,bin,oct
3	data segment
4	infor db 0ah,0dh,'1.dec to bin'
5	db 0ah,0dh,'2.dec to oct'
6	db 0ah,0dh,'3.dec to hex'
7	db 0ah,0dh,'4.exit'
8	db 0ah,0dh,'select:\$'
9	mess0 db 0ah,0dh,'error.reinput,please.',0ah,0dh,\$'
10	mess1 db 0ah,0dh,'input dec(0-255)=\$'
11	mess2 db 0ah,0dh,'output bin=\$'
12	mess3 db 0ah,0dh,'output oct=0\$'
13	mess4 db 0ah,0dh,'output hex=0x0\$'
14	mess5 db 0ah,0dh,'input dec(0-65535)=\$'
15	x1 dw 0
16	x db 3 dup(?) ;16 进制的
17	bin db '01'
18	oct db '01234567'
19	data ends
20	stack segment para stack 'stack'
21	dw 20 dup(0)
22	stack ends
23	code segment para 'code'
24	assume cs:code,ds:data,ss:data
25	start:
26	mov ax,data
27	mov ds,ax
28	main proc far
29	mov dx,offset infor
30	mov ah,9
31	int 21h
32	mov ah,1
33	int 21h
34	call prog1
35	call prog2

行号	5-1.asm 代码
36	call prog3
37	call prog4
38	jmp main
39	main endp
40	code ends
41	end start

表 2. 5-2.asm 代码

行号	5-2.asm 代码
1	public prog1
2	extrn mess0:byte,mess2:byte,mess5:byte,x1:word,bin:byte
3	code segment para 'code'
4	assume cs:code
5	prog1 proc far ;10 to 2
6	cmp al,'1'
7	jnz exit
8	let0:
9	mov x1,0
10	mov dx,offset mess5
11	mov ah,9
12	int 21h ;显示输入提示
13	let1:
14	mov ah,1
15	int 21h
16	cmp al,27
17	jz re
18	sub al,30h
19	jl let2
20	cmp al,9
21	jg let2
22	mov ah,0
23	xchg ax,x1
24	mov cx,10
25	mul cx
26	xchg ax,x1
27	add x1,ax

行号	5-2.asm 代码
28	jc reinput ;结果进位则转移
29	jmp let1
30	exit:
31	ret
32	re: ;初始化 al,使其能 jmp main
33	mov al,0
34	ret
35	reinput:
36	mov dx,offset mess0 ;reinput
37	mov ah,9
38	int 21h
39	jmp let0
40	let2:
41	mov dx,offset mess2
42	mov ah,9
43	int 21h ;显示输出提示
44	mov bx,x1
45	mov ch,16 ;共有 16 位数
46	mov cl,1 ;每次循环左移 1 位
47	let3:
48	rol bx,cl
49	mov al,bl
50	and ax,0001h ;保留最低 1 位
51	mov di,ax
52	mov dl,bin[di]
53	mov ah,2
54	int 21h
55	dec ch
56	jnz let3
57	jmp let0
58	prog1 endp
59	code ends
60	end

表 3. 5-3.asm 代码

行号	5-3.asm 代码
----	------------

行号	5-3.asm 代码
1	public prog2
2	extrn mess0:byte,mess3:byte,mess5:byte,x1:word,oct:byte
3	code segment para 'code'
4	assume cs:code
5	prog2 proc far ;10 to 8
6	cmp al,'2'
7	jnz exit
8	let0:
9	mov x1,0
10	mov dx,offset mess5
11	mov ah,9
12	int 21h ;显示输入提示
13	let1:
14	mov ah,1
15	int 21h
16	cmp al,27
17	jz re
18	sub al,30h
19	jl let2
20	cmp al,9
21	jg let2
22	mov ah,0
23	xchg ax,x1
24	mov cx,10
25	mul cx
26	xchg ax,x1
27	add x1,ax
28	jc reinput ;结果进位则转移
29	jmp let1
30	exit:
31	ret
32	re: ;初始化 al,使其能 jmp main
33	mov al,0
34	ret
35	reinput:
36	mov dx,offset mess0 ;reinput
37	mov ah,9
38	int 21h

行号	5-3.asm 代码
39	jmp let0
40	let2:
41	mov dx,offset mess3
42	mov ah,9
43	int 21h ;显示输出提示
44	mov bx,x1
45	mov ch,6 ;共有 6 位数
46	let3:
47	mov cl,1 ;循环左移 1 位
48	rol bx,cl
49	mov al,bl
50	and ax,0001h ;保留最低 1 位
51	mov di,ax
52	mov dl,oct[di]
53	mov ah,2
54	int 21h
55	dec ch
56	let4:
57	mov cl,3 ;循环左移 3 位
58	rol bx,cl
59	mov al,bl
60	and ax,0007h ;保留最低 3 位
61	mov di,ax
62	mov dl,oct[di]
63	mov ah,2
64	int 21h
65	dec ch
66	jnz let4
67	jmp let0
68	prog2 endp
69	code ends
70	end

表 4. 5-4.asm 代码

行号	5-4.asm 代码
1	public prog3

行号	5-4.asm 代码
2	extrn mess0:byte,mess1:byte,mess4:byte,x:byte
3	code segment para 'code'
4	assume cs:code
5	prog3 proc far ;10 to 16
6	cmp al,'3'
7	jnz exit
8	let0:
9	mov x,0
10	mov x+1,0
11	mov x+2,0
12	mov si,0
13	mov dx,offset mess1
14	mov ah,9
15	int 21h ;显示输入提示
16	let1:
17	mov ah,1
18	int 21h ;输入十进制数字
19	cmp al,27 ;是否 ESC
20	jz re
21	cmp al,0dh ;是否回车 CR
22	jz let2
23	and ax,000fh;去掉 ASCII 码
24	mov x[si],al ;保存到 x
25	inc si ;si 自加 1
26	jmp let1
27	exit:
28	ret
29	re: ;初始化 al,使其能 jmp main
30	mov al,0
31	ret
32	let2:
33	cmp si,2
34	ja let6
35	cmp si,1 ;判断输入的位数
36	ja let3 ;输入两位转向 let33
37	mov bl,x
38	mov cl,1
39	mov dx,offset mess4

行号	5-4.asm 代码
40	mov ah,9
41	int 21h ;显示输出提示
42	jmp let5 ;只输入一位数则直接去显示
43	let6:
44	mov al,x
45	mov cl,10
46	mul cl ;形成两位十进制数
47	add al,x+1
48	mov cl,10
49	mul cl ;形成三位十进制数（255 以内）
50	add al,x+2
51	jc let7 ;结果进位 cf=1 时 reinput
52	cmp ax,00ffh
53	ja let7 ;结果高于 ffh 时 reinput
54	mov bl,16 ;除以 16，转换为 hex
55	div bl
56	mov bx,ax ;ah 为余数是低位，al 为商是高位
57	;分别显示 hex 的高位和低位
58	mov cl,2
59	mov dx,offset mess4
60	mov ah,9
61	int 21h ;显示输出提示
62	jmp let4
63	let7:
64	mov dx,offset mess0 ;reinput
65	mov ah,9
66	int 21h
67	jmp let0
68	let3:
69	mov al,x
70	mov cl,10
71	mul cl ;形成两位十进制数
72	add al,x+1
73	mov ah,0
74	mov bl,16 ;除以 16，转换为 hex
75	div bl
76	mov bx,ax ;ah 为余数是低位，al 为商是高位
77	;分别显示 hex 的高位和低位

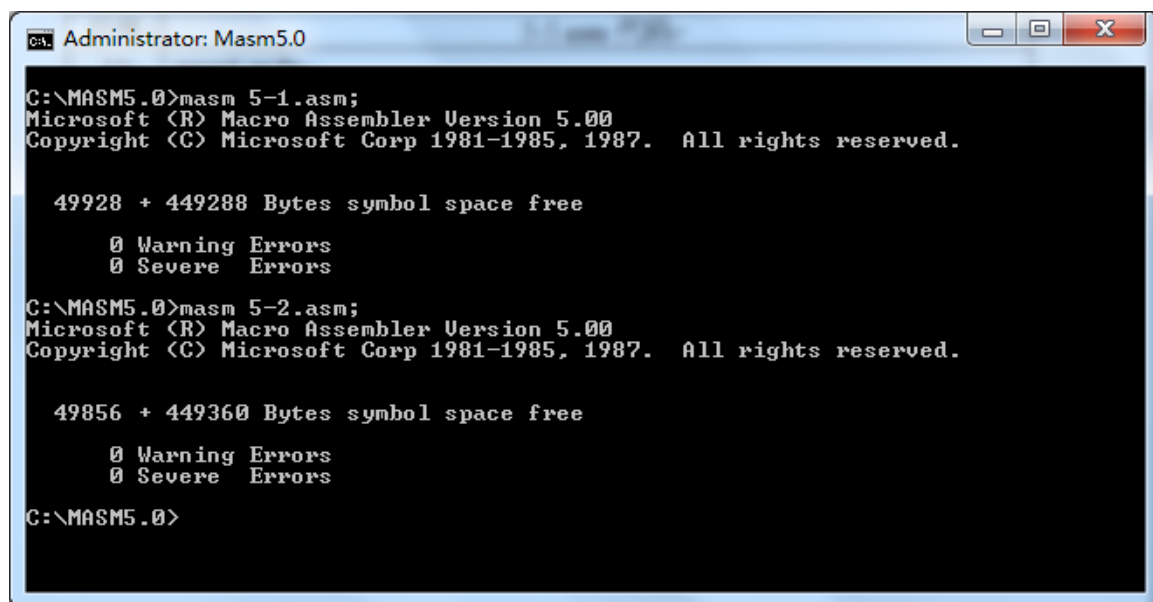
行号	5-4.asm 代码
78	mov cl,2
79	mov dx,offset mess4
80	mov ah,9
81	int 21h ;显示输出提示
82	let4:
83	cmp bl,10 ;判断 hex 数码
84	jl let5
85	add bl,7 ;>10 则加 7, A-F
86	let5:
87	add bl,30h
88	mov dl,bl
89	mov ah,2
90	int 21h ;显示
91	mov bl,bh ;再去显示低位
92	dec cl
93	jnz let4
94	jmp let0
95	prog3 endp
96	code ends
97	end

表 5. 5-5.asm 代码

行号	5-5.asm 代码
1	public prog4
2	code segment para 'code'
3	assume cs:code
4	prog4 proc far ;exit
5	cmp al,27
6	jz let1
7	cmp al,'4'
8	jnz exit
9	let1:
10	mov ah,4ch
11	int 21h
12	exit:
13	ret

行号	5-5.asm 代码
14	prog4 endp
15	code ends
16	end

二、实验结果



```

Administrator: Masm5.0

C:\MASM5.0>masm 5-1.asm;
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

49928 + 449288 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\MASM5.0>masm 5-2.asm;
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

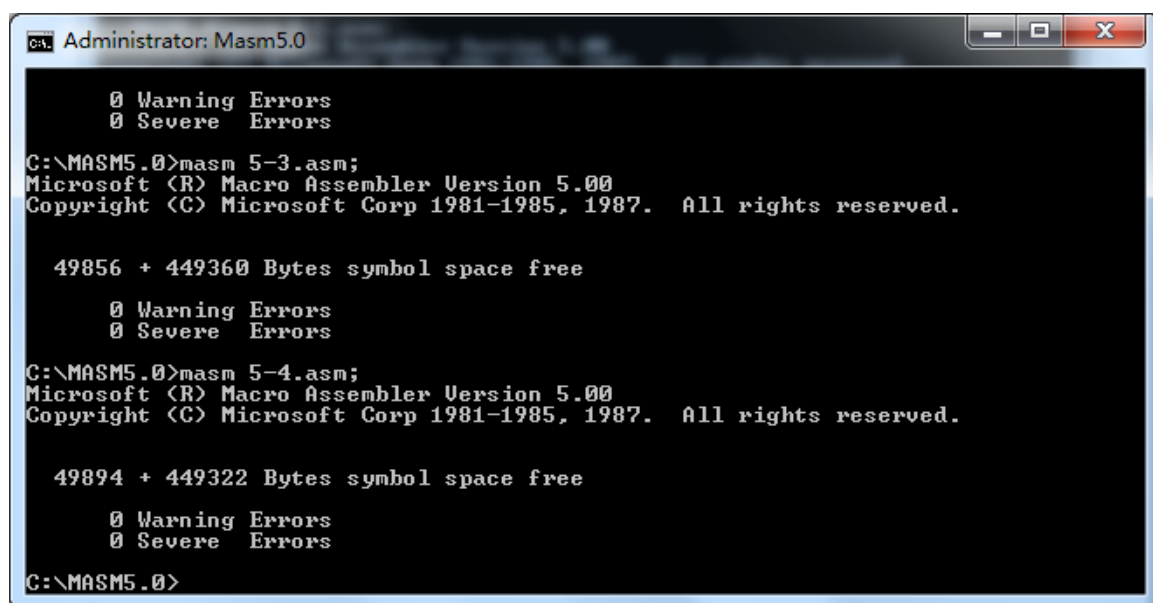
49856 + 449360 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\MASM5.0>

```

图 1. 将 5-1.asm 和 5-2.asm 进行汇编



```

Administrator: Masm5.0

0 Warning Errors
0 Severe Errors

C:\MASM5.0>masm 5-3.asm;
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

49856 + 449360 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\MASM5.0>masm 5-4.asm;
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

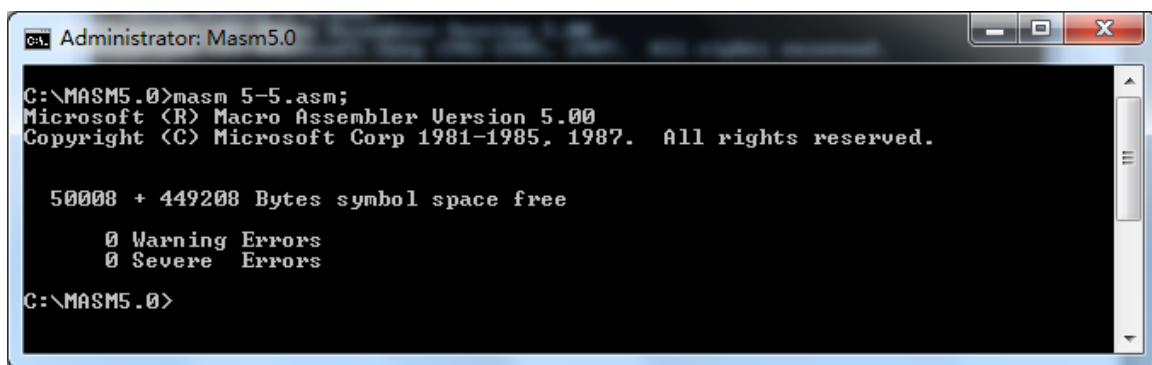
49894 + 449322 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\MASM5.0>

```

图 2. 将 5-3.asm 和 5-4.asm 进行汇编



```
Administrator: Masm5.0

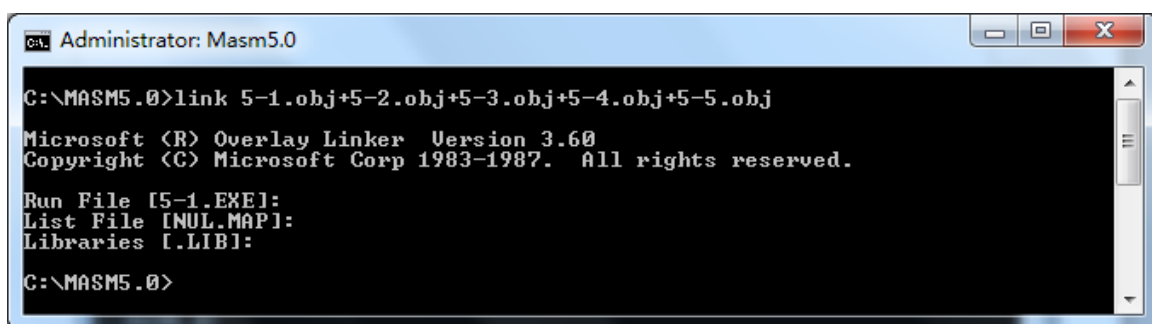
C:\MASM5.0>masm 5-5.asm;
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

50000 + 449208 Bytes symbol space free

      0 Warning Errors
      0 Severe Errors

C:\MASM5.0>
```

图 3. 将 5-5.asm 进行汇编



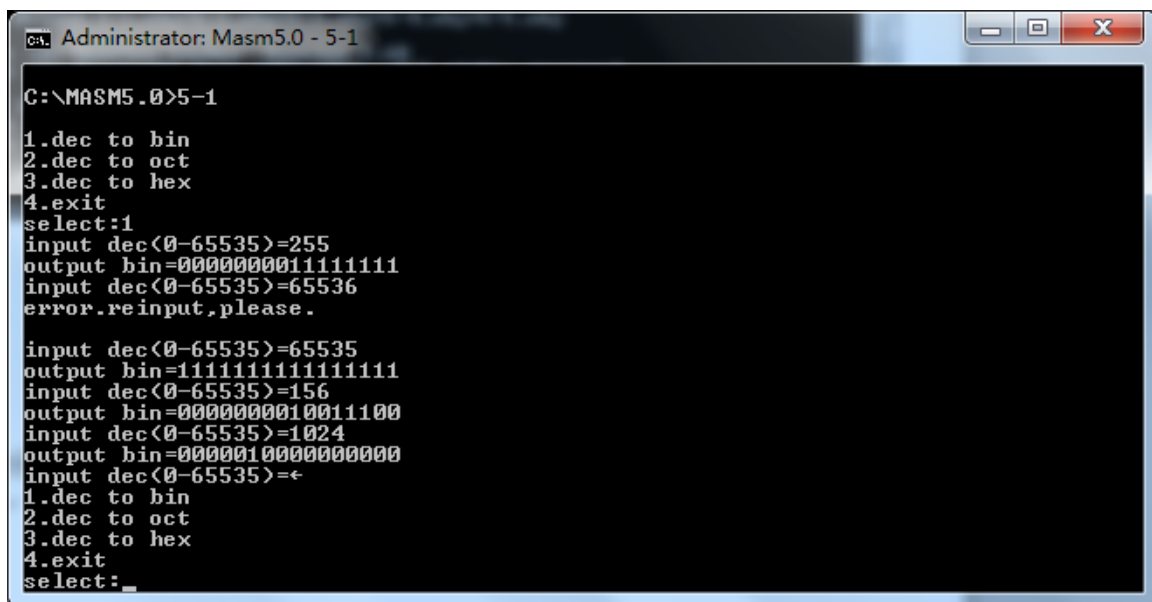
```
Administrator: Masm5.0

C:\MASM5.0>link 5-1.obj+5-2.obj+5-3.obj+5-4.obj+5-5.obj
Microsoft (R) Overlay Linker Version 3.60
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.

Run File [5-1.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:

C:\MASM5.0>
```

图 4. 用 link 命令将 5 个 OBJ 文件连接



```
Administrator: Masm5.0 - 5-1

C:\MASM5.0>5-1

1.dec to bin
2.dec to oct
3.dec to hex
4.exit
select:1
input dec(0-65535)=255
output bin=000000001111111
input dec(0-65535)=65536
error.reinput,please.

input dec(0-65535)=65535
output bin=111111111111111
input dec(0-65535)=156
output bin=000000010011100
input dec(0-65535)=1024
output bin=0000010000000000
input dec(0-65535)=←
1.dec to bin
2.dec to oct
3.dec to hex
4.exit
select:_
```

图 5. 十到二进制转换测试

```
Administrator: Masm5.0 - 5-1

1.dec to bin
2.dec to oct
3.dec to hex
4.exit
select:2
input dec(0-65535)=256
output oct=0000400
input dec(0-65535)=211
output oct=0000323
input dec(0-65535)=65536
error.reinput,please.

input dec(0-65535)=65535
output oct=0177777
input dec(0-65535)=255
output oct=0000377
input dec(0-65535)=8
output oct=0000010
input dec(0-65535)=←
1.dec to bin
2.dec to oct
3.dec to hex
4.exit
select:
```

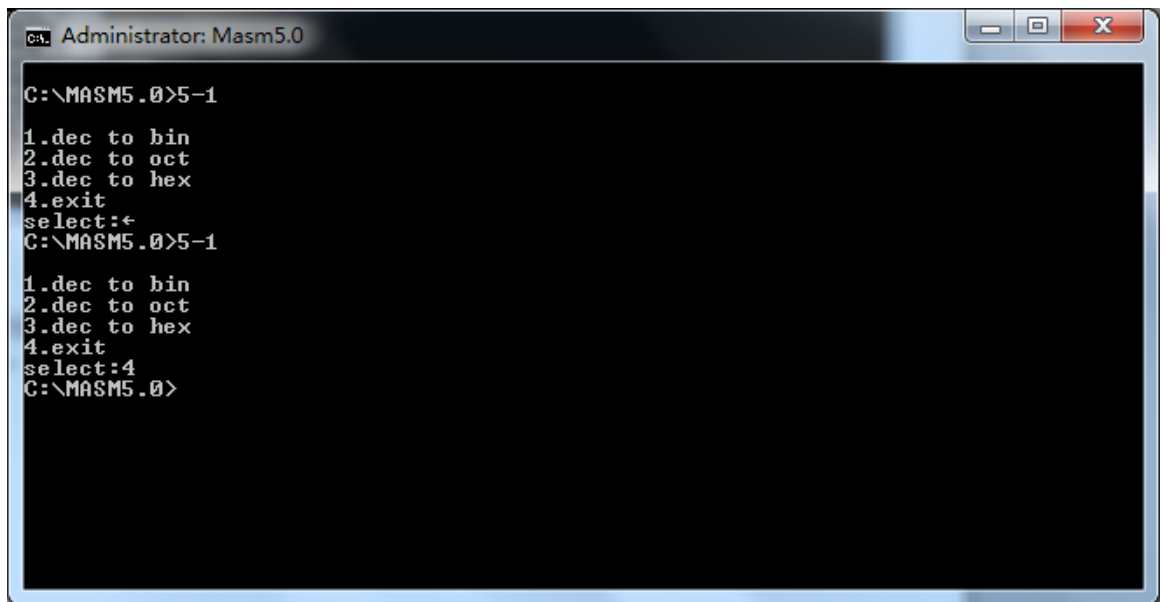
图 6. 十到八进制转换测试

```
Administrator: Masm5.0 - 5-1

1.dec to bin
2.dec to oct
3.dec to hex
4.exit
select:3
input dec(0-255)=255
output hex=0xFF
input dec(0-255)=256
error.reinput,please.

input dec(0-255)=16
output hex=0x010
input dec(0-255)=10
output hex=0x00A
input dec(0-255)=13
output hex=0x00D
input dec(0-255)=128
output hex=0x080
input dec(0-255)=←
1.dec to bin
2.dec to oct
3.dec to hex
4.exit
select: _
```

图 7. 十到十六进制转换测试



```
C:\MASM5.0>5-1
1.dec to bin
2.dec to oct
3.dec to hex
4.exit
select:←
C:\MASM5.0>5-1
1.dec to bin
2.dec to oct
3.dec to hex
4.exit
select:4
C:\MASM5.0>
```

图 4. 退出功能测试

三、实验分析

这个程序分别转换了十进制到二进制、十进制到八进制、十进制到十六进制。设计一个主程序 main，四个子程序 prog1、prog2、prog3、prog4，用五个模块分别保存，实现远程的访问与调用。主程序的功能是调用四个子程序。prog1 的功能是 65535 以内十进制转换二进制。prog2 的功能是 65535 以内十进制转换八进制。prog3 的功能是 255 以内十进制转换十六进制。prog4 的功能是判断并退出程序。通过对五个 asm 文件进行汇编并 link 方式连接，实现模块化的程序设计。