

2.6 mov al, 89h; al = 89h OF = PF = SF = ZF = CF = 0  
 add al, al; al = 12h OF = CF = PF = 1 SF = ZF = 0  
 add al, 9dh; al = af SF = PF = 1 CF = ZF = OF = 0  
 cmp al, 00ch; al = af SF = PF = CF = 1 ZF = OF = 0  
 sub al, al; al = 0 PF = ZF = 1 SF = OF = CF = 0  
 dec al; al = ff SF = PF = 1 ZF = CF = OF = 0  
 inc al; al = 0 PF = ZF = 1 SF = CF = OF = 0

- 2.7 (1) add dx, bx  
 (2) add al, [bx + 6i]  
 (3) add cx, [bx + 02h]  
 (4) add word ptr [0520], 3412  
 (5) add al, 0a0

2.1 解: (1) ax = 1200

(2) ax = 100

(3) ax = 4C2A

(4) ax = 3412

(5) ax = 4C2A

(6) ax = 7856

(7) ax = 65B7

2.3 (1) bx ← table 有效地址 ea, bx = 200H

(2) al ← r

(3) al ← ds:[bx + al] al = 121H

2.5

mov ax, 8057h; ax ← 8057h

push ax; sp ← 2, 2200:0000 ← sp

mov ax, 0779h; ax ← 0779h

push ax; sp ← 2, 2200:0000 ← sp

pop bx; bx ← 0779, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

pop [bx]; ds:[0779] ← 8057h, sp ← 2

2.2 (1) dl 改为 dx

(2) ip 不能作为操作数

(3) 不能将直接数 mov 至段寄存器

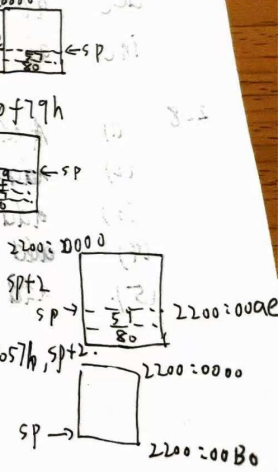
(4) 两个操作数不能全是段寄存器

(5) mov al 的源小于 255

(6) 寄存器间接寻址只能用 BX, BP, SI, DI

(7) 缺少方括号

目的 (6) 操作数不能是立即数



2.13

(1)

```
DATAS SEGMENT
    ;此处输入数据段代码
DATAS ENDS

STACKS SEGMENT
    ;此处输入堆栈段代码
STACKS ENDS

CODES SEGMENT
    ASSUME CS:CODES,DS:DATAS,SS:STACKS
START:
    MOV AX,DATAS
    MOV DS,AX
    xor ah,ah;将ah清空
    mov al,10
    mov bx,ax
    mov cl,3
    shl ax,cl
    add ax,bx
    add ax,bx
    MOV AH,4CH
    INT 21H
CODES ENDS
    END START
```

表 1 t2.13 (1)

(2)

01 DATAS SEGMENT

;此处输入数据段代码

03 DATAS ENDS

04

05 STACKS SEGMENT

;此处输入堆栈段代码

07 STACKS ENDS

08

09 CODES SEGMENT

10 ASSUME CS:CODES,DS:DATAS,SS:STACKS

11 START:

12 MOV AX,DATAS

13 MOV DS,AX

14 mov ax,'0'

15 sub al,30h

16 MOV AH,4CH

17 INT 21H

18 CODES ENDS

19 END START

20

DOSBox 0.74, Cpu speed: 3000 cycles, Frames...

-r

AX=FFFF BX=0000 CX=000E DX=0000 SP=0000 BP=0000 SI=0000 DI=0000

DS=0760 ES=0760 SS=076F CS=0770 IP=0000 NU UP EI PL NZ NA PO NC

0770:0000 B87007

MOV AX,0770

-t

AX=0770 BX=0000 CX=000E DX=0000 SP=0000 BP=0000 SI=0000 DI=0000

DS=0760 ES=0760 SS=076F CS=0770 IP=0003 NU UP EI PL NZ NA PO NC

0770:0003 8ED8

MOV DS,AX

-t

AX=0770 BX=0000 CX=000E DX=0000 SP=0000 BP=0000 SI=0000 DI=0000

DS=0770 ES=0760 SS=076F CS=0770 IP=0005 NU UP EI PL NZ NA PO NC

0770:0005 B83000

MOV AX,0030

-t

AX=0030 BX=0000 CX=000E DX=0000 SP=0000 BP=0000 SI=0000 DI=0000

DS=0770 ES=0760 SS=076F CS=0770 IP=0008 NU UP EI PL NZ NA PO NC

0770:0008 2C30

SUB AL,30

-t

AX=0000 BX=0000 CX=000E DX=0000 SP=0000 BP=0000 SI=0000 DI=0000

DS=0770 ES=0760 SS=076F CS=0770 IP=000A NU UP EI PL NZ NA PE NC

0770:000A B44C

MOV AH,4C

-t

表 2 t2.13 (2)

(3)

01 DATAS SEGMENT

;此处输入数据段代码

03 DATAS ENDS

04

05 STACKS SEGMENT

;此处输入堆栈段代码

07 STACKS ENDS

08

09 CODES SEGMENT

10 ASSUME CS:CODES,DS:DATAS,SS:STACKS

11 START:

12 MOV AX,DATAS

13 MOV DS,AX

14 mov cx,4 ; 设置循环次数为 4

15 loop1:

16 shr dx,1 ; 将 ax 右移一位

17 rcr ax,1 ; 将 dx 右移一位,同时将 ax 的最高位移入 dx 的最低位

18 loop loop1

19 MOV AH,4CH

20 INT 21H

21 CODES ENDS

22 END START

23

表 3 2.13 (3)

2.14

```

01 DATAS SEGMENT
02     ;此处输入数据段代码
03 DATAS ENDS
04
05 STACKS SEGMENT
06     ;此处输入堆栈段代码
07 STACKS ENDS
08
09 CODES SEGMENT
10     ASSUME CS:CODES,DS:DATAS,SS:STACKS
11 START:
12     MOV AX,DATAS
13     MOV DS,AX
14     mov al, 0F7h
15     sar al, 1;商是-5,fb是-5
16
17     MOV AH,4CH
18     INT 21H
19 CODES ENDS
20     END START
21

```

DOSBox 0.74, Cpu speed: 3000 cycles, Frames...

AX=FFFF	BX=0000	CX=0000	DX=0000	SP=0000	BP=0000	SI=0000	DI=0000
DS=0760	ES=0760	SS=076F	CS=0770	IP=0000	NU	UP	EI
0770:0000	B87007	MOV	AX,0770				
-t							
AX=0770	BX=0000	CX=0000	DX=0000	SP=0000	BP=0000	SI=0000	DI=0000
DS=0760	ES=0760	SS=076F	CS=0770	IP=0003	NU	UP	EI
0770:0003	BED0	MOV	DS,AX				
-t							
AX=0770	BX=0000	CX=0000	DX=0000	SP=0000	BP=0000	SI=0000	DI=0000
DS=0770	ES=0760	SS=076F	CS=0770	IP=0005	NU	UP	EI
0770:0005	B0F7	MOV	AL,F7				
-t							
AX=07F7	BX=0000	CX=0000	DX=0000	SP=0000	BP=0000	SI=0000	DI=0000
DS=0770	ES=0760	SS=076F	CS=0770	IP=0007	NU	UP	EI
0770:0007	D0F8	SAR	AL,1				
-t							
AX=07FB	BX=0000	CX=0000	DX=0000	SP=0000	BP=0000	SI=0000	DI=0000
DS=0770	ES=0760	SS=076F	CS=0770	IP=0009	NU	UP	EI
0770:0009	B44C	MOV	AH,4C				

```

DATAS SEGMENT
    ;此处输入数据段代码
DATAS ENDS

STACKS SEGMENT
    ;此处输入堆栈段代码
STACKS ENDS

CODES SEGMENT
    ASSUME CS:CODES,DS:DATAS,SS:STACKS
START:
    MOV AX,DATAS
    MOV DS,AX
    ;mov al, 0F7h
    ;sar al, 1;商是-5,fb是-5
    mov bl, 2
    mov al, 0F7h
    mov ah, 0
    idiv bl ;商是123,7b是123,余数是1
    MOV AH,4CH
    INT 21H
CODES ENDS
    END START

```

DOSBox 0.74, Cpu speed: 3000 cycles, Frames...

AX=0770	BX=0000	CX=0011	DX=0000	SP=0000	BP=0000	SI=0000	DI=0000
DS=0770	ES=0760	SS=076F	CS=0770	IP=0005	NU	UP	EI
0770:0005	B302	MOV	BL,02				
-t							
AX=0770	BX=0002	CX=0011	DX=0000	SP=0000	BP=0000	SI=0000	DI=0000
DS=0770	ES=0760	SS=076F	CS=0770	IP=0007	NU	UP	EI
0770:0007	B0F7	MOV	AL,F7				
-t							
AX=07F7	BX=0002	CX=0011	DX=0000	SP=0000	BP=0000	SI=0000	DI=0000
DS=0770	ES=0760	SS=076F	CS=0770	IP=0009	NU	UP	EI
0770:0009	B400	MOV	AH,00				
-t							
AX=00F7	BX=0002	CX=0011	DX=0000	SP=0000	BP=0000	SI=0000	DI=0000
DS=0770	ES=0760	SS=076F	CS=0770	IP=000B	NU	UP	EI
0770:000B	F6FB	IDIV	BL				
-t							
AX=017B	BX=0002	CX=0011	DX=0000	SP=0000	BP=0000	SI=0000	DI=0000
DS=0770	ES=0760	SS=076F	CS=0770	IP=000D	NU	UP	EI
0770:000D	B44C	MOV	AH,4C				