

## Experiment 5 Pseudo-instructions

### Purpose:

Master the common pseudo-instructions to write a complete program, implement the following data definition statements and related instructions, view the definition of memory data under Debug, and single-step trace to view the value of each register.

### Experimental content:

1. Write a complete program with 16-bit instructions, implement the following data definition statements and related instructions, check the definition of memory data under Debug, and check the values of each register with single-step tracking.

```
DATA SEGMENT
ORG 0
ARRAY LABEL BYTE
DA1 DW 2, 9, 14, 3, 315H, -6
DA2 DB 7, 'ABCDEF'
LEN = $-DA2
ORG 100H
DA3 DW DA4
DA4 DB 4 DUP(2 DUP(1, 2, 3), 4)
DATA ENDS
```

```
CODE SEGMENT
ASSUME CS:CODE, DS:DATA
START:
MOV AX, DATA
MOV DX, AX
MOV AL, ARRAY+2
ADD AL, DA2+1
MOV AX, DA2-DA1
MOV BL, LEN
MOV AX, DA3
MOV BX, TYPE DA4
MOV BX, OFFSET DA4
MOV CX, SIZE DA4
MOV DX, LENGTH DA4
MOV BX, WORD PTR DA4
MOV BL, LEN AND 0FH
MOV BL, LEN GT 5
MOV AX, LEN MOD 5

MOV AH, 4CH
INT 21H
```

CODE ENDS

END START

```
DOSBox 0.74, Cpu speed: 3000 cycles, Fra...
LINK : warning L4021: no stack segment

C:\>debug h2.exe
-u
077C:0000 BB6A07      MOV     AX,076A
077C:0003 BBD0        MOV     DX,AX
077C:0005 A00200      MOV     AL,[0002]
077C:0008 02060D00    ADD     AL,[000D]
077C:000C BB0C00      MOV     AX,000C
077C:000F B309        MOV     BL,09
077C:0011 A10001      MOV     AX,[0100]
077C:0014 BB0100      MOV     BX,0001
077C:0017 BB0201      MOV     BX,0102
077C:001A B90400      MOV     CX,0004
077C:001D BA0400      MOV     DX,0004
-d 076a:0
076A:0000 02 00 09 00 0E 00 03 00-15 03 FA FF 07 41 42 43 .....ABC
076A:0010 44 45 44 46 47 00 00 00-00 00 00 00 00 00 00 00 DEDFG.....
076A:0020 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
076A:0030 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
076A:0040 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
076A:0050 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
076A:0060 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
076A:0070 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
```

```
DOSBox 0.74, Cpu speed: 3000 cycles, Fra...
AX=076A BX=0000 CX=014F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=0003 NU UP EI PL NZ NA PO NC
077C:0003 BBD0      MOV     DX,AX
-t
AX=076A BX=0000 CX=014F DX=076A SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=0005 NU UP EI PL NZ NA PO NC
077C:0005 A00200      MOV     AL,[0002]          DS:0002=FF
-t
AX=07FF BX=0000 CX=014F DX=076A SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=0008 NU UP EI PL NZ NA PO NC
077C:0008 02060D00    ADD     AL,[000D]          DS:000D=01
-t
AX=0700 BX=0000 CX=014F DX=076A SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=000C NU UP EI PL ZR AC PE CY
077C:000C BB0C00      MOV     AX,000C
-t
AX=000C BX=0000 CX=014F DX=076A SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=000F NU UP EI PL ZR AC PE CY
077C:000F B309        MOV     BL,09
```

```

DOSBox 0.74, Cpu speed: 3000 cycles, Fra...
AX=0000 BX=0009 CX=014F DX=076A SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=0011 NU UP EI PL ZR AC PE CY
077C:0011 A10001 MOV AX,[0100] DS:0100=0002
-t
AX=0002 BX=0009 CX=014F DX=076A SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=0014 NU UP EI PL ZR AC PE CY
077C:0014 BB0100 MOV BX,0001
-t
AX=0002 BX=0001 CX=014F DX=076A SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=0017 NU UP EI PL ZR AC PE CY
077C:0017 BB0201 MOV BX,0102
-t
AX=0002 BX=0102 CX=014F DX=076A SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=001A NU UP EI PL ZR AC PE CY
077C:001A B90400 MOV CX,0004
-t
AX=0002 BX=0102 CX=0004 DX=076A SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=001D NU UP EI PL ZR AC PE CY
077C:001D BA0400 MOV DX,0004
-

```

```

DOSBox 0.74, Cpu speed: 3000 cycles, Fra...
AX=0002 BX=0102 CX=0004 DX=0004 SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=0020 NU UP EI PL ZR AC PE CY
077C:0020 BB1E0201 MOV BX,[0102] DS:0102=0009
-t
AX=0002 BX=0009 CX=0004 DX=0004 SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=0024 NU UP EI PL ZR AC PE CY
077C:0024 B309 MOV BL,09
-t
AX=0002 BX=0009 CX=0004 DX=0004 SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=0026 NU UP EI PL ZR AC PE CY
077C:0026 B3FF MOV BL,FF
-t
AX=0002 BX=00FF CX=0004 DX=0004 SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=0028 NU UP EI PL ZR AC PE CY
077C:0028 BB0400 MOV AX,0004
-t
AX=0004 BX=00FF CX=0004 DX=0004 SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=077C IP=002B NU UP EI PL ZR AC PE CY
077C:002B B44C MOV AH,4C
-

```

2、 Can the following program output 0~9? If not, how should it be modified?

CODE SEGMENT

ASSUME CS:CODE

K=30H

J DW 0

START:MOV DL,K

MOV AH,2

INT 21H

K=K+1

INC J

CMP J,10

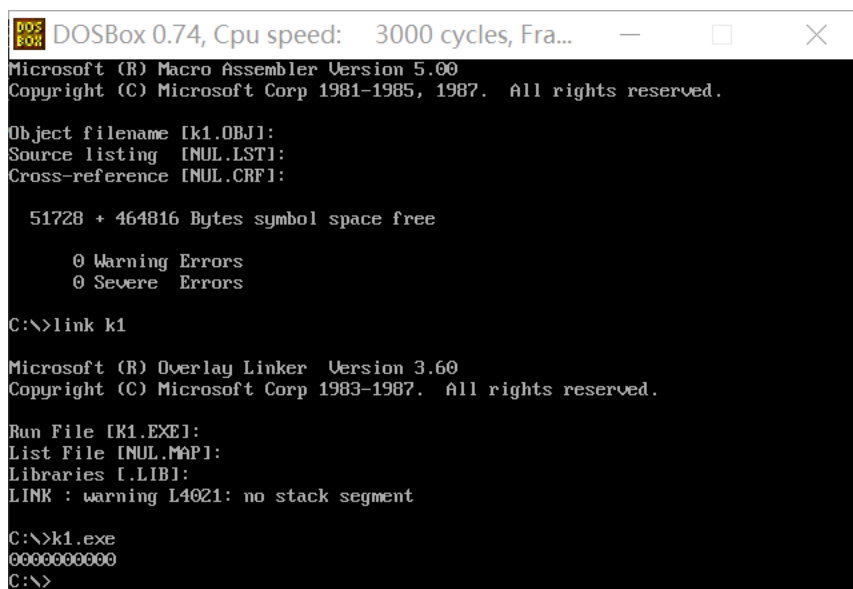
JNZ START

MOV AH,4CH

INT 21H

CODE ENDS

END START



```
DOSBox 0.74, Cpu speed: 3000 cycles, Fra...
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

Object filename [k1.OBJ]:
Source listing [NUL.LST]:
Cross-reference [NUL.CRF]:

51728 + 464816 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\>link k1

Microsoft (R) Overlay Linker Version 3.60
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.

Run File [K1.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:
LINK : warning L4021: no stack segment

C:\>k1.exe
0000000000
C:\>
```

answer:

As shown in the picture, no .

The following modifications should be made:

Change the third line [ K=30H ] to [ K DB 30H ]

Change line 8 [ K=K+1 ] to [ INC K ]

The modified code is as follows:

CODE SEGMENT

ASSUME CS:CODE

K DB 30H

J DW 0

START:MOV DL, K

```

MOV AH, 2
INT 21H
INC K
INC J
CMP J, 10
JNZ START
MOV AH, 4CH
INT 21H
CODE ENDS
END START

```

```

DOSBox 0.74, Cpu speed: 3000 cycles, Fra...
LINK : warning L4021: no stack segment
C:\>h1.exe
0123456789
C:\>debug h1.exe
-u
076A:0003 2E          CS:
076A:0004 8A160000        MOV     DL,[0000]
076A:0008 B402          MOV     AH,02
076A:000A CD21          INT     21
076A:000C 2E          CS:
076A:000D FE060000        INC     BYTE PTR [0000]
076A:0011 2E          CS:
076A:0012 FF060100        INC     WORD PTR [0001]
076A:0016 2E          CS:
076A:0017 833E01000A      CMP     WORD PTR [0001],+0A
076A:001C 75E5          JNZ     0003
076A:001E B44C          MOV     AH,4C
076A:0020 CD21          INT     21
076A:0022 83C404        ADD     SP,+04
-q
C:\>h1.exe
0123456789
C:\>

```

As shown in the figure, the display outputs 0~9.