第一次实验报告

一、实验目的

1. 熟悉 DOSBOX 环境和常用指令
2. 熟悉常用寻址方式

二、设计说明

1. DOSBOX 使用方法

```
Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>set path=c:\masm;c:\tasm

Drive C is mounted as local directory c:\Users\Lucid-X\.vscode\extensions\xsro.m
asm-tasm-0.9.0\tools\

Drive D is mounted as local directory d:\VSCode\VS-Code-Assmbler\

Drive X is mounted as local directory c:\Users\Lucid-X\AppData\Roaming\Code\User
\globalStorage\xsro.masm-tasm\
```

(1) 工作环境

(2) 建立汇编文件

①使用汇编工具生成 obj 文件

```
D:\>masm 1.asm
Microsoft (R) MASM Compatibility Driver
Copyright (C) Microsoft Corp 1993.  All rights reserved.

 Invoking: ML.EXE /I. /Zm /c /Ta 1.asm

Microsoft (R) Macro Assembler Version 6.11
Copyright (C) Microsoft Corp 1981-1993.  All rights reserved.
```

②使用链接工具生成 exe 文件

```
Run File [1.exe]:
List File [nul.map]:
Libraries [.lib]:
Definitions File [nul.def]:
1.asm : fatal error L1101: invalid object module
Object file offset: 1 Record type: 44
```

③执行 exe 文件

```
Microsoft (R) Segmented Executable Linker  Version 5.31.009 Jul 13 1992
Copyright (C) Microsoft Corp 1984-1992.  All rights reserved.

Run File [1.exe]:  a
```

(3) 常用指令

①T：逐指令跟踪程序命令（TRACE）

从指定地址执行一条指令后停止，显示寄存器内存储以及标志位的值

未指定地址从 CS：IP 开始执行

```
-t

AX=2000  BX=0000  CX=0016  DX=0000  SP=0000  BP=0000  SI=0000  DI=0000
DS=075C  ES=075C  SS=076B  CS=076C  IP=0003   NV UP EI PL NZ NA PO NC
076C:0003 8ED0          MOV     SS,AX
```

②U: 反汇编指令（Unassemble）

对指定内存区域的内容以汇编语言显示，同时显示地址和对应的机器码

```
-u
0740:0103 0000          ADD     [BX+SI],AL
0740:0105 0000          ADD     [BX+SI],AL
0740:0107 0000          ADD     [BX+SI],AL
0740:0109 0000          ADD     [BX+SI],AL
0740:010B 0000          ADD     [BX+SI],AL
0740:010D 00AEFE00      ADD     [BP+00FE],CH
0740:0111 F0            LOCK
0740:0112 46            INC     SI
0740:0113 7400          JZ      0115
0740:0115 00B200B2      ADD     [BP+SI+B200],DH
0740:0119 16            PUSH    SS
0740:011A 99            CWD
0740:011B 002F          ADD     [BX],CH
0740:011D 07            POP     ES
0740:011E 2F            DAS
0740:011F 07            POP     ES
0740:0120 0000          ADD     [BX+SI],AL
0740:0122 0000          ADD     [BX+SI],AL
-_▲
```

③A: 汇编指令（assemble）

允许键入汇编语言语句，并将其汇编成机器代码，依次存储

```
-a
0740:0100 mov ax,0002
0740:0103
-r
AX=0000  BX=0000  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=0100   NV UP EI PL NZ NA PO NC
0740:0100 B80200        MOV     AX,0002
-t

AX=0002  BX=0000  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=0103   NV UP EI PL NZ NA PO NC
0740:0103 0000          ADD     [BX+SI],AL                      DS:0000=CD
```

以回车结束，并显示下一行语句的起始地址。

Debug 状态下键入的数字都是做十六进制。

在未指明段地址的前提下，以 CS 的值作为段地址。

④D: 显示存储单元（Dump）

功能：以两种形式显示指定内存范围的内容。

十六进制显示和 ASCII 显示（不可显示的字符用“.”代替）

```
-d
0740:0100  B8 02 00 00 00 00 00 00-00 00 00 00 00 00 AE FE   ................
0740:0110  00 F0 46 74 00 00 B2 00-B2 16 99 00 2F 07 2F 07   ..Ft.......././.
0740:0120  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0740:0130  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0740:0140  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0740:0150  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0740:0160  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
0740:0170  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
```

⑤E:修改内存单元内容（Enter）

```
-e ds:100
0740:0100  B8.77
```

显示指定存储单元的内容，并等待用户键入新的值

可通过空格连续键入

⑥F：填写内存单元（FILL）

```
-F 4BA:0100 L5 F3'XYZ'8D
-D 4BA:0100
04BA:0100  F3 58 59 5A 8D 61 6C 6C-6F 63 61 74 65 20 65 78   .XYZ.allocate ex
04BA:0110  70 61 6E 64 65 64 20 6D-65 6D 6F 72 79 20 20 20   panded memory
04BA:0120  20 20 20 20 20 58 41 20-5B 23 70 61 67 65 73 5D        XA [#pages]
04BA:0130  0D 0A 64 65 61 6C 6C 6F-63 61 74 65 20 65 78 70   ..deallocate exp
04BA:0140  61 6E 64 65 64 20 6D 65-6D 6F 72 79 20 20 20 20   anded memory
04BA:0150  20 20 58 44 20 5B 68 61-6E 64 6C 65 5D 0D 0A 6D     XD [handle]..m
04BA:0160  61 70 20 65 78 70 61 6E-64 65 64 20 6D 65 6D 6F   ap expanded memo
04BA:0170  72 79 20 70 61 67 65 73-20 20 20 20 20 20 20 58   ry pages       X
```

将 List 中的内容逐字填入指定内存单元，List 用完后自动重复使用，需要指定长度

⑦P：逐行追踪程序指令（Proceed）

用于结束 CALL 和 INT N 指令。

⑧Q: 退出 Debug 指令

用于推出 Debug

⑨R：检查和修改寄存器内容的指令（Register）

显示所有寄存器内容和状态位

```
-R
AX=0000  BX=0000  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=0100   NV UP EI PL NZ NA PO NC
0740:0100 0000          ADD     [BX+SI],AL                      DS:0000=CD
```

显示和修改某个寄存器的内容

修改则输入新的值，不修改直接 enter

```
-R AX
AX 0000
:7777
-R
AX=7777  BX=0000  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=0100   NV UP EI PL NZ NA PO NC
0740:0100 0000          ADD     [BX+SI],AL                      DS:0000=CD
```

（4）MS-DOS 方式命令

①DIR 命令

显示当前目录下的文件文件和目录

```
D:\>DIR
Directory of D:\.
.              <DIR>          22-10-2021  0:11
..             <DIR>          06-10-2021 21:49
1       ASM                266 21-10-2021 22:25
1       MAP                195 22-10-2021  0:11
    2 File(s)              461 Bytes.
    2 Dir(s)       262,111,744 Bytes free.
```

②CD 命令

改变当前工作目录

③MD 命令

新建一个工作目录

④DEL 命令

删除文件

⑤RD 命令

删除目录，要求删除目录下没有其他的文件

⑥COPY 命令

复制文件

2. 寻址方式

（1）立即数寻址

MOV AX,1000

```
D:\>DEBUG
-A 100
0740:0100 MOV AX,1000
0740:0103
-T=0100

AX=1000  BX=0000  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=0103   NV UP EI PL NZ NA PO NC
0740:0103 0000          ADD     [BX+SI],AL                      DS:0000=CD
```

MOV AL,BB

```
0740:0103 MOV AL,BB
0740:0105
-T=0103

AX=10BB  BX=0000  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=0105   NV UP EI PL NZ NA PO NC
0740:0105 0000          ADD     [BX+SI],AL                      DS:0000=CD
```

通过寄存器指明操作类型，AX 说明是字操作，AL 说明是字节操作

EA=(IDATA) SA=(DS);

将立即数存放在内存代码段中，在 CPU 取指令的时候随指令码一起取出参与运算

（2）直接寻址

MOV AX,1200

MOV [1000],AX

MOV BX,[1000]

```
D:\>DEBUG
-A 0100
0740:0100 MOV AX,1200
0740:0103 MOV [1000],AX
0740:0106 MOV BX,[1000]
0740:010A
-T=0100

AX=1200  BX=0000  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=0103   NV UP EI PL NZ NA PO NC
0740:0103 A30010        MOV     [1000],AX                       DS:1000=0000
-T

AX=1200  BX=0000  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=0106   NV UP EI PL NZ NA PO NC
0740:0106 8B1E0010      MOV     BX,[1000]                       DS:1000=1200
-T

AX=1200  BX=1200  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=010A   NV UP EI PL NZ NA PO NC
0740:010A 0000          ADD     [BX+SI],AL                      DS:1200=00
```

未指明段地址则默认为 DS，[]内用十六位常数存放数据的偏移地址

可通过指明段地址的方式改变默认段地址 MOV AX,ES:[BX]

此时段地址为 ES

（3）间接寻址

MOV AX,1200

MOV [1000],AX

MOV AX,0000

MOV BX,1000

```
AX=1200  BX=1200  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=0103   NV UP EI PL NZ NA PO NC
0740:0103 A30010        MOV     [1000],AX                       DS:1000=1200
-T

AX=1200  BX=1200  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=0106   NV UP EI PL NZ NA PO NC
0740:0106 B80000        MOV     AX,0000
-T

AX=0000  BX=1200  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=0109   NV UP EI PL NZ NA PO NC
0740:0109 BB0010        MOV     BX,1000
-T

AX=0000  BX=1000  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=010C   NV UP EI PL NZ NA PO NC
0740:010C 8B07          MOV     AX,[BX]                         DS:1000=1200
- ▲
```

MOV AX,[BX]

允许作为操作数偏移地址的寄存器只有 SI、DI、BX、BP，称为间址寄存器或地址指针.

不同的间址寄存器对应的段寄存器不同，BX,SI,DI 对应 DS，BP 对应 SS.

[BX]　　EA=(BX),SA=(DS)

[SI]　　EA=(SI),SA=(DS)

[DI]　　EA=(DI),SA=(DS)

[BP]　　EA=(BP),SA=(SS)

（4）相对寻址

MOV AX,1200

MOV [1002],AX

MOV AX,0000

MOV BX,1000

MOV AX,2[BX]

```
AX=1200  BX=1000  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=0103   NV UP EI PL NZ NA PO NC
0740:0103 A30210        MOV     [1002],AX                       DS:1002=0000
-T

AX=1200  BX=1000  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=0106   NV UP EI PL NZ NA PO NC
0740:0106 B80000        MOV     AX,0000
-T

AX=0000  BX=1000  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=0109   NV UP EI PL NZ NA PO NC
0740:0109 BB0010        MOV     BX,1000
-T

AX=0000  BX=1000  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=0740  ES=0740  SS=0740  CS=0740  IP=010C   NV UP EI PL NZ NA PO NC
0740:010C 8B4702        MOV     AX,[BX+02]                      DS:1002=1200
```

[BX+IDATA]　　EA=(BX+IDATA),SA=(DS)

[SI+IDATA]　　EA=(SI+IDATA),SA=(DS)

[DI+IDATA]　　EA=(DI+IDATA),SA=(BX)

[BP+IDATA]　　EA=(BP+IDATA),SA=(SS)

不同的间址寄存器对应的段寄存器不同，BX,SI,DI 对应 DS，BP 对应 SS.

（5）基址-变址寻址

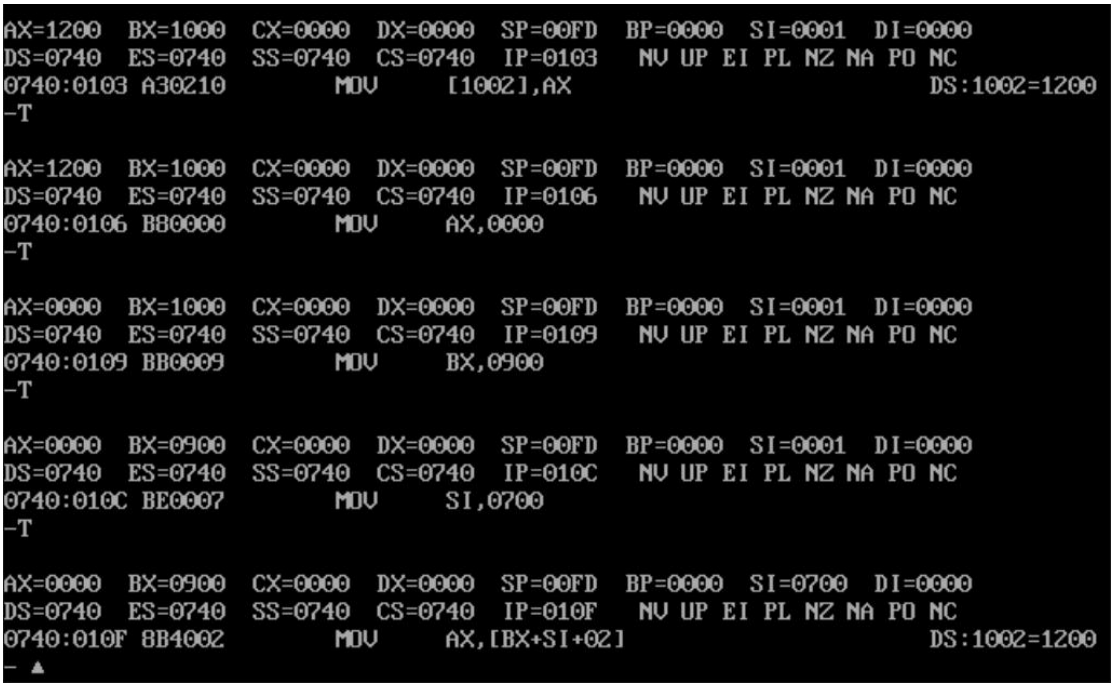
MOV AX,1200

MOV [1002],AX

MOV AX,0000

MOV BX,900

MOV SI,700



MOV AX,2[BX][SI]

[BX+SI+IDATA]　　EA=(BX)+(SI)+IDATA,SA=(DS)

[BX+DI+IDATA]　　EA=(BX)+(DI)+IDATA,SA=(DS)

[BP+SI+IDATA]　　EA=(BP)+(SI)+IDATA,SA=(SS)

[BP+DI+IDATA]　　EA=(BP)+(DI)+IDATA,SA=(SS)

只允许一个基址寄存器+一个变址寄存器，其中 BX 对应 DS，BP 对应 SS.

三、心得体会

熟悉了 DOS 环境的基本操作方式，熟悉了寻址方式，了解了代码在内存中是如何运行的。