华 中 科 技 大 学

课 程 实 验 报 告

课程名称：汇编语言程序设计实验

实验名称：汇编语言程序设计课程实验一

实验时间：2019.09.16

实验地点：南一楼116\_1

指导教师：鲁宏伟

专业班级：信安1801

学 号：U201814834

姓 名：杨傲

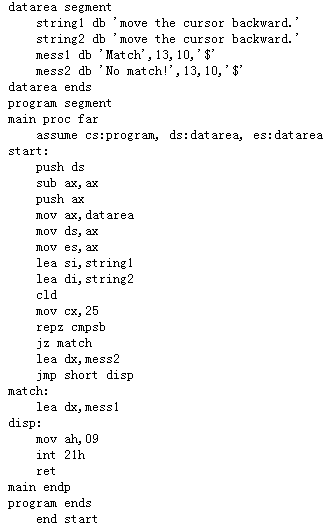
报告日期：2019.09.16

**成绩评定**

|  |  |  |
| --- | --- | --- |
| 实验完成质量（70分） | 报告撰写质量（30分） | 总成绩 |
| 实验步骤清晰、详细、深入，实验记录真实完整等 | 报告规范、完整、通顺、详实 |
|  |  |  |

1. 实验目的与要求
2. 掌握汇编源程序编辑工具、汇编程序、连接程序、调试工具 TD 的使用；
3. 理解数、符号、寻址方式等在计算机内的表现形式；
4. 理解指令执行与标志位改变之间的关系；
5. 熟悉常用的 DOS 功能调用；
6. 熟悉分支、循环程序的结构及控制方法，掌握分支、循环程序的调试方法；
7. 加深对转移指令及一些常用的汇编指令的理解。
8. 实验内容

任务 1. 阅读以下程序，根据指令的执行流程，说明程序实现的目标，并采用汇编程序对改代码进行编辑、编译、连接和调试。



要求：

1. 查阅资料，说明指令“cld”和“repz cmpsb”的功能；
2. 查阅资料，说明“int 21h”的功能；
3. 借助TD工具，观察指令“repz cmpsb”完成后，哪些标志位发生了变化。
4. 程序中的两条“push”指令的作用是什么，有没有感觉代码中缺少与“push”指令相关的操作指令？如果有，尝试完善该段代码，并观察程序运行效果。
5. 运行程序，说明当前程序运行的结果，然后修改字符串string2，让程序产生不同的结果。

任务 2. 阅读下列程序, 并指出程序执行之后, 以BUF2、BUF3、BUF4 为首址的3 个字节存储区中存放的数据。

|  |  |
| --- | --- |
|  |  |

要求：

1. 分别记录执行到“ MOV CX，10”和“INT 21H”之前的(BX)、(BP)、(SI)、(DI)各是多少。
2. 记录程序执行到退出之前数据段开始40个字节的内容，指出程序运行结果是否与设想的一致。
3. 在标号LOPA前加上一段程序，实现新的功能：先显示提示信息“Press any key to begin!”, 然后，在按了一个键之后继续执行 LOPA 处的程序。

操作提示：使用 TD.EXE 调试程序时，应先单步执行各个语句，每执行一条语句，都应观察数据段中的内容以及相应寄存器的变化。首先注意观察对DS寄存器的赋值过程，并在TD的数据窗口定位待观察的数据区位置。其次，单步执行循环体两遍且正确理解了循环体语句的含义后，可在“MOV AH, 4CH”处设置断点，然后直接执行到断点处，回答(1)和(2)的问题。完成（3）的内容，涉及到“INT 21H”相关的输入和输出操作，参考任务1中显示字符串的方法。

1. 实验过程

在进行实验任务前先搭建汇编语言开发环境，使用老师提供的Masm6.15\_downcc.zip解压出含有masm必要程序的文件夹，利用DOSBox0.74-win32-installer.zip解压出DosBox的安装程序进行DosBox在电脑上的安装，之后利用TD.zip解压出491808TD.EXE，将之放入masm文件夹中，根据TD调试工具使用手册进行TD的使用。

在DosBox安装完成后，进入其默认安装位置找到DOSBox 0.74 Options.bat，在文件末尾部分添上

mount c d:\huibian

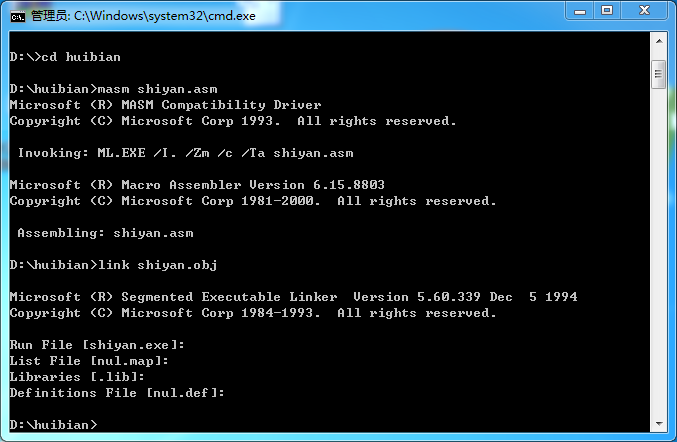
set path=$path$;d:\huibian

c:

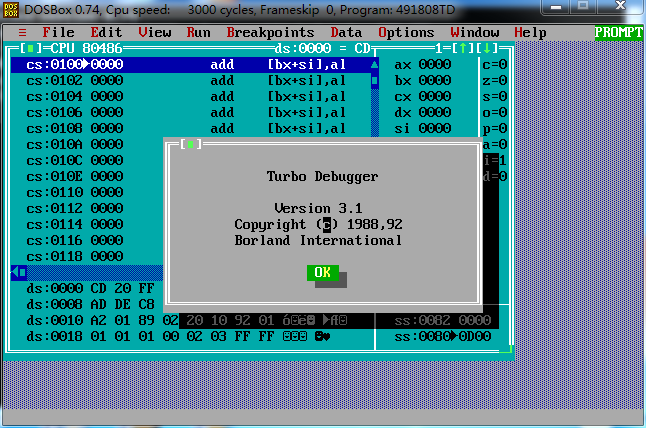
之后再打开DosBox进行使用时便可以自动转到此目录下，节省一点精力。之后的任务中，使用电脑自带文本文档进行编辑源代码，之后修改文件格式为asm文件即可（当然也可以使用masm文件夹中的EDIT.exe进行编辑），得到asm文件后，放入masm文件夹中，启动电脑cmd.exe，在其中对asm文件进行汇编生成obj文件。



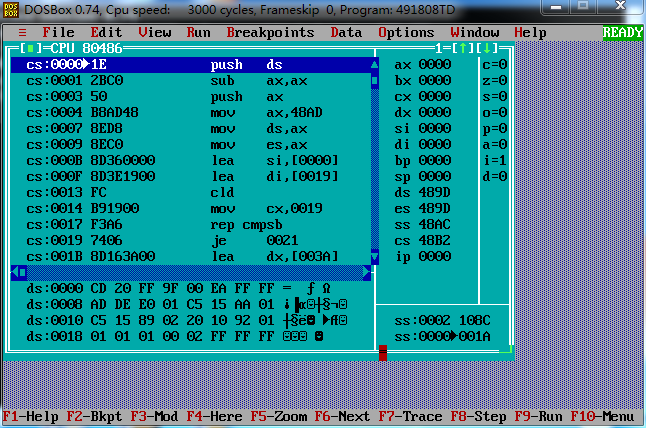
之后再对生成的obj文件进行链接，生成对应的exe文件。



启动DosBox，打开TD，



再在TD中打开生成的exe文件，即可开始调试。



任务一

**解答:**

程序实现的目标是检验string1和string2是否相同，相同则输出mess1中的信息‘Match’，即相符；不同则输出mess2中的信息‘No match！’，即不相符。

（1）

cld指令功能：

将标志寄存器的方向标志位DF清零。 在字串操作中使变址寄存器SI或DI的地址指针自动增加，字串处理由前往后。

repz cmpsb功能：

repe是一个串操作前缀，它重复串操作指令，每重复一次ECX的值就减一，一直到CX为0或ZF为0时停止。 cmpsb是字符串比较指令，把ESI指向的数据与EDI指向的数一个一个的进行比较。当repe cmpsb配合使用时就是字符串比较，当相同时继续比较，不同时不比较，即如果ds:si和es:di所指向的两个字节相等，则继续比较。所以如果不相等的话，就停止循环。

（2）

int 21h功能：

int 21h是DOS功能调用，调用功能号放在AH中，使用格式如下：

MOV AH,01

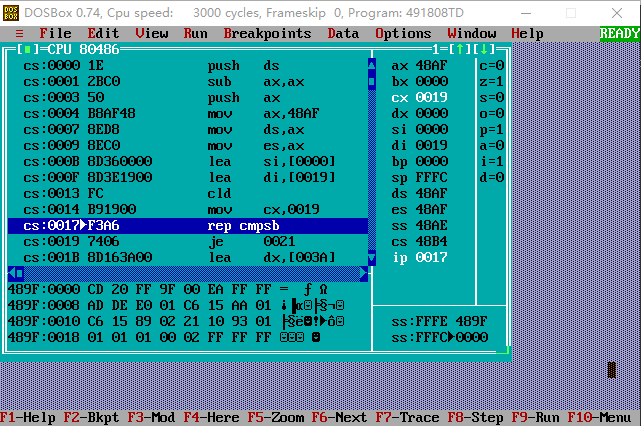
INT 21H

此时AH中存储有01的调用功能号，而01对应的功能为键盘输入并回显，调用参数无，返回参数为AL=输入字符。功能号有很多种，网上有表格。

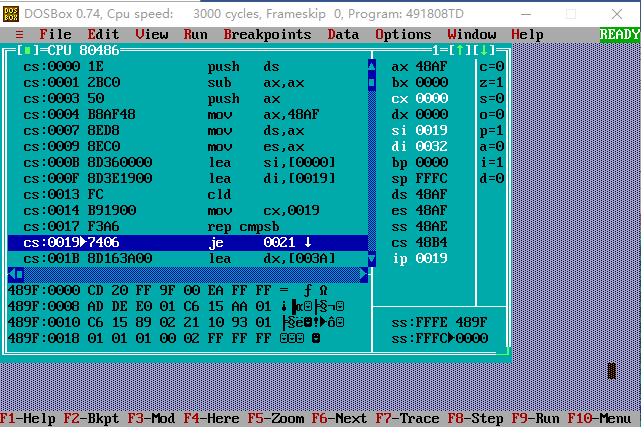
（3）

借助TD工具，观察指令“repz cmpsb”完成后，哪些标志位发生了变化。

repz cmpsb这一步执行前：



这一步执行后：



在repz cmpsb这一步中未发生标志位变化。

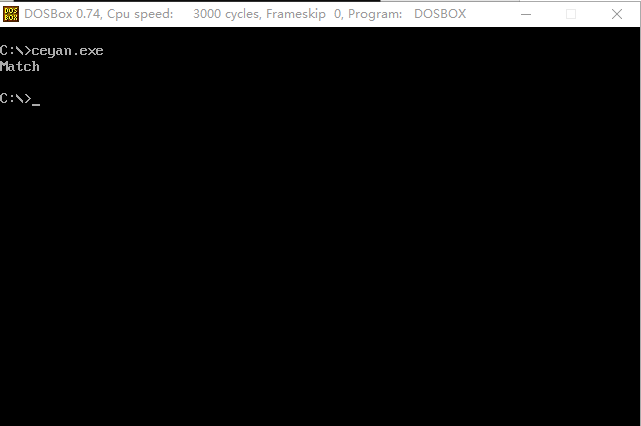
（4）程序中的两条“push”指令的作用是什么，有没有感觉代码中缺少与“push”指令相关的操作指令？如果有，尝试完善该段代码，并观察程序运行效果。

push指令用来将寄存器中的值保存到栈中。此处用来存储返回地址，配合ret退出程序。而ret指令用栈中的数据，修改IP的值，从而实现近转移。

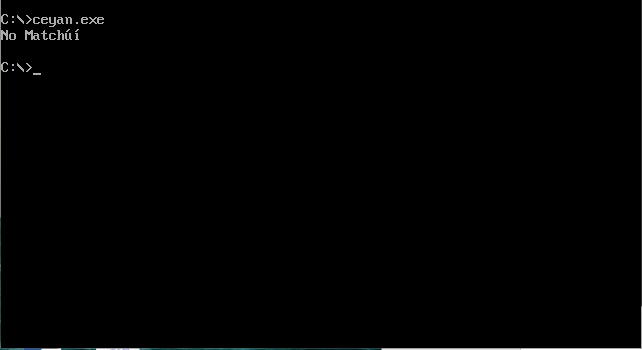
CPU执行ret指令时，进行下面两步操作：(IP)=((SS)\*16+(SP))和(SP)=(SP)+2;

此处ret指令相当于pop ip; pop cs;之前将ds和0入栈，执行到ret的时候将cs:ip获得栈中数据变为ds:0，而ds:0即int 20h指令，可以调用int 20h回dos。

（5）结果为‘Match’



修改string2使string1和string2不同后结果为‘No match！’。



任务二

**解答：**

**题干回答：**

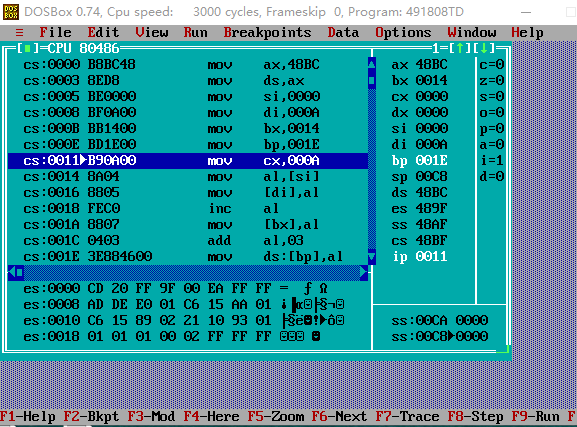
**BUF2 00 01 02**

**BUF3 01 02 03**

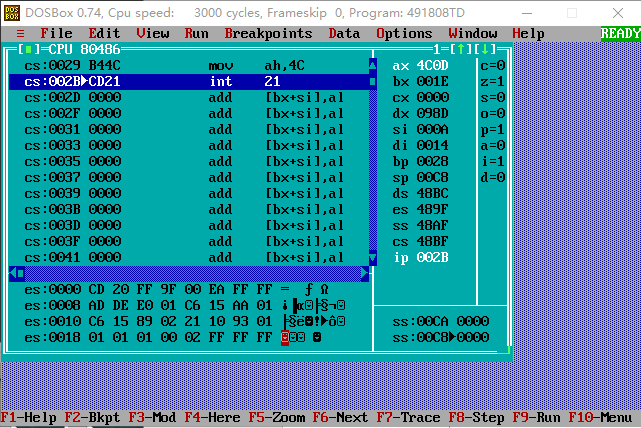
**BUF4 04 05 06**

（1）

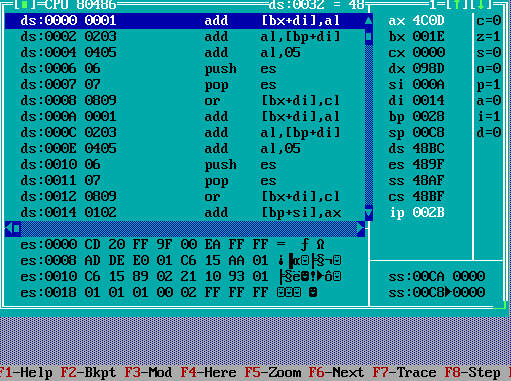
执行到“ MOV CX，10”前： bx 0014 bp 001E si 0000 di 000A

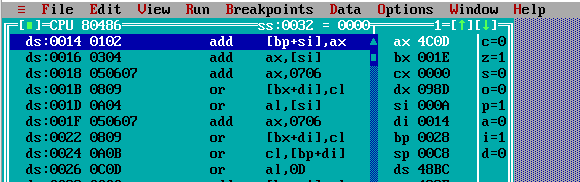


执行到 “INT 21H”之前：bx 001E bp 0028 si 000A di 0014



（2）记录程序执行到退出之前数据段开始40个字节的内容，指出程序运行结果是否与设想的一致。





BUF1给到了BUF2

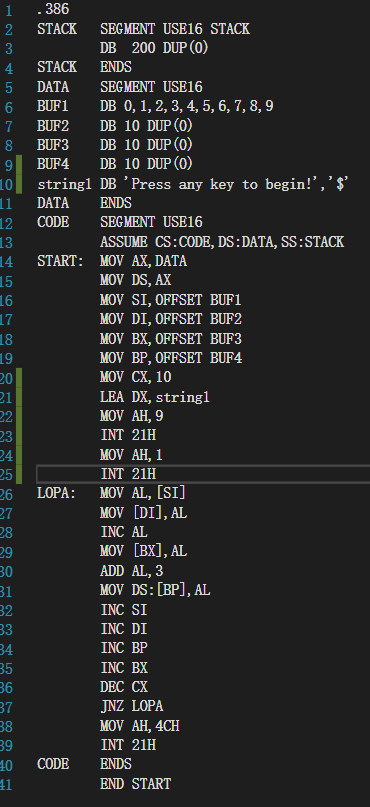
BUF1值加1后给到了BUF3

BUF1值加4后给到了BUF4

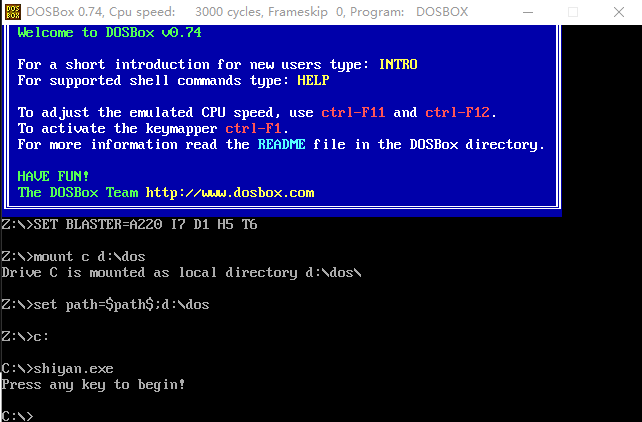
和预想的一致。

（3）在标号LOPA前加上一段程序，实现新的功能：先显示提示信息“Press any key to begin!”, 然后，在按了一个键之后继续执行 LOPA 处的程序。

修改后：



此时，汇编、链接后运行程序



操作提示：使用 TD.EXE 调试程序时，应先单步执行各个语句，每执行一条语句，都应观察数据段中的内容以及相应寄存器的变化。首先注意观察对DS寄存器的赋值过程，并在TD的数据窗口定位待观察的数据区位置。其次，单步执行循环体两遍且正确理解了循环体语句的含义后，可在“MOV AH, 4CH”处设置断点，然后直接执行到断点处，回答(1)和(2)的问题。完成（3）的内容，涉及到“INT 21H”相关的输入和输出操作，参考任务1中显示字符串的方法。

1. 总结与体会

在本次课程中，学习了如何编写汇编程序，并对其进行汇编、链接生成exe文件，以及如何利用DosBox和TD在电脑上调试汇编程序完成任务。另外，在完成任务的过程中，学习了int 21h的调用知识。

加深了对于汇编语言中循环、字符串等知识的理解，自学了repz cmpsb指令，cld指令的用法。本次实验中尤其在自己通过查找资料解决问题的过程中，收获到了很多意料之外的知识，加深了对于汇编语言的理解，让我感到十分有收获和惊喜。