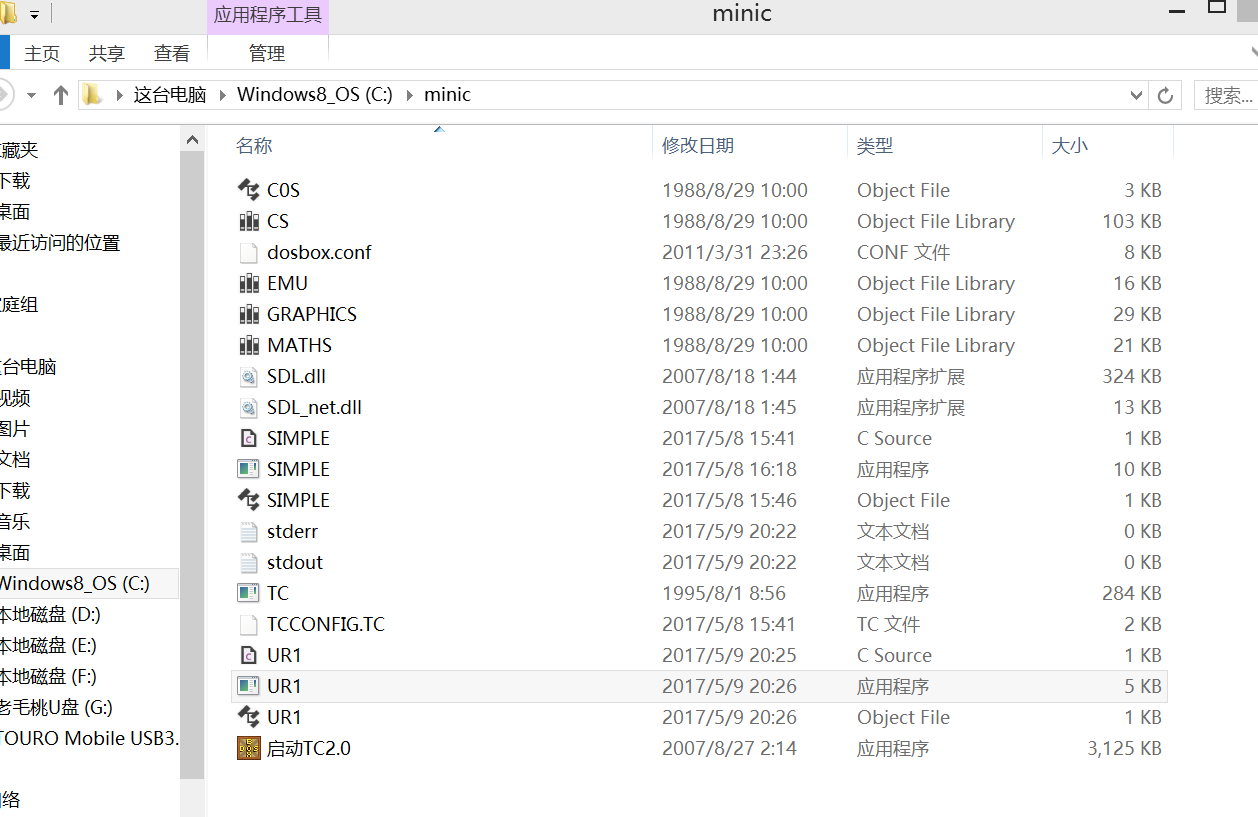
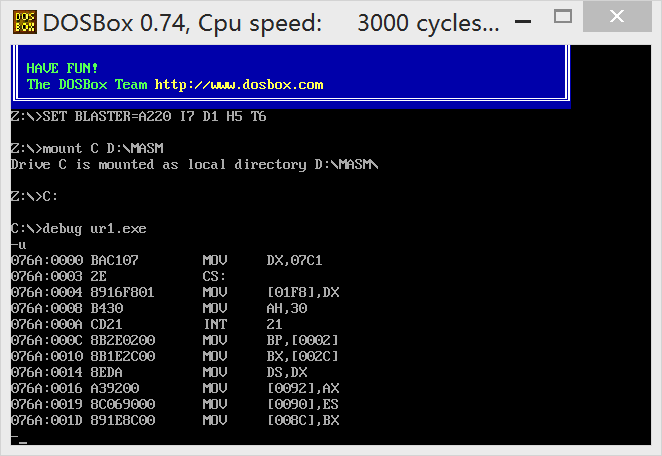
**综合研究二研究报告**

1. **研究过程展示**

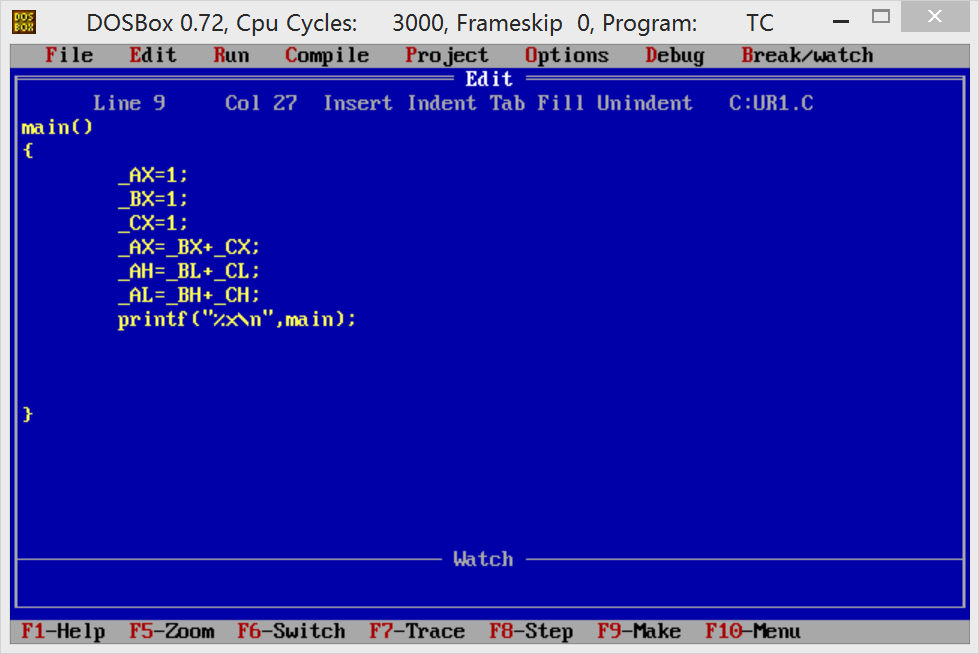
在综合研究一的基础上，很快地编译连接生成了UR1.EXE文件。如图所示：

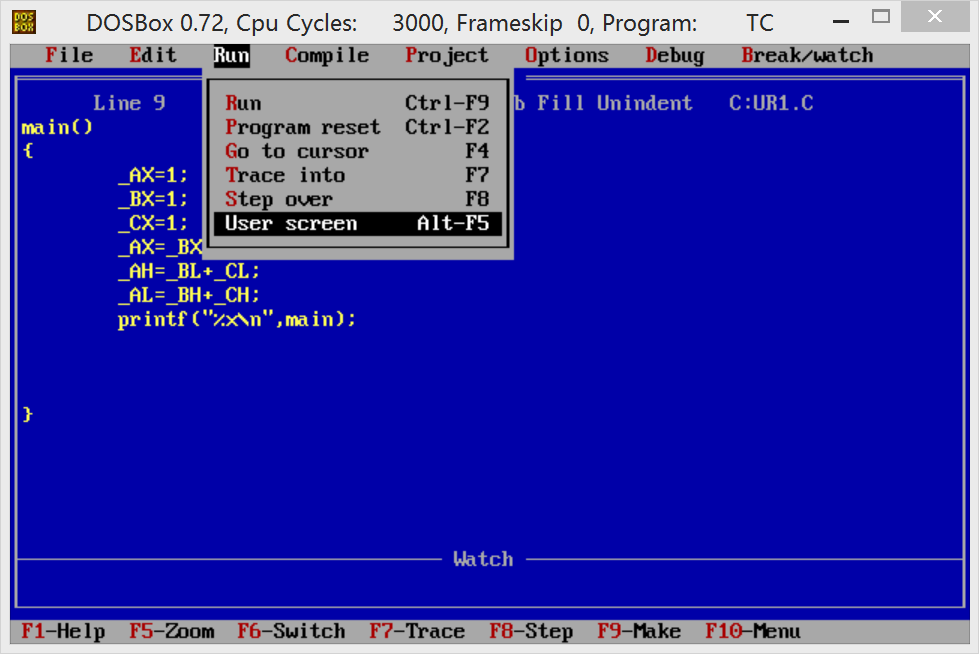
****

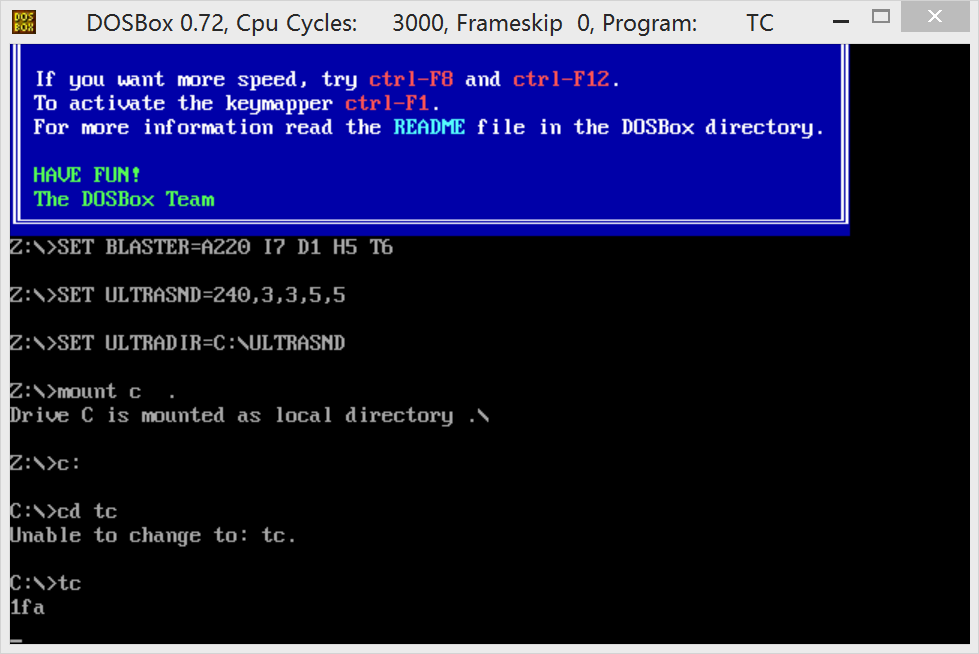
用Debug加载UR1.exe，用u命令查看UR1.c编译之后的机器码和汇编代码，但是用u命令之后，出现的反汇编代码却不是预想的那样，没有一处符合，有很大可能这不是main函数存放的位置，或者我应该向前或者向后移一点？

****

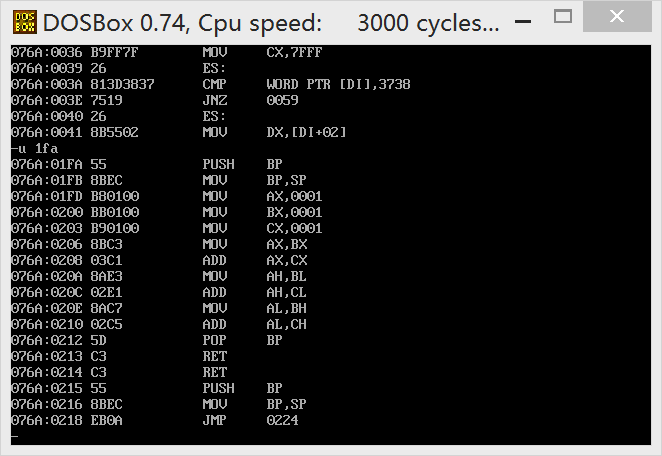
不过好在书上在提出这个问题的时候，也提出一个建议，试着做一下：

****

****

****

再u一下，这样明显很符合预期：

****

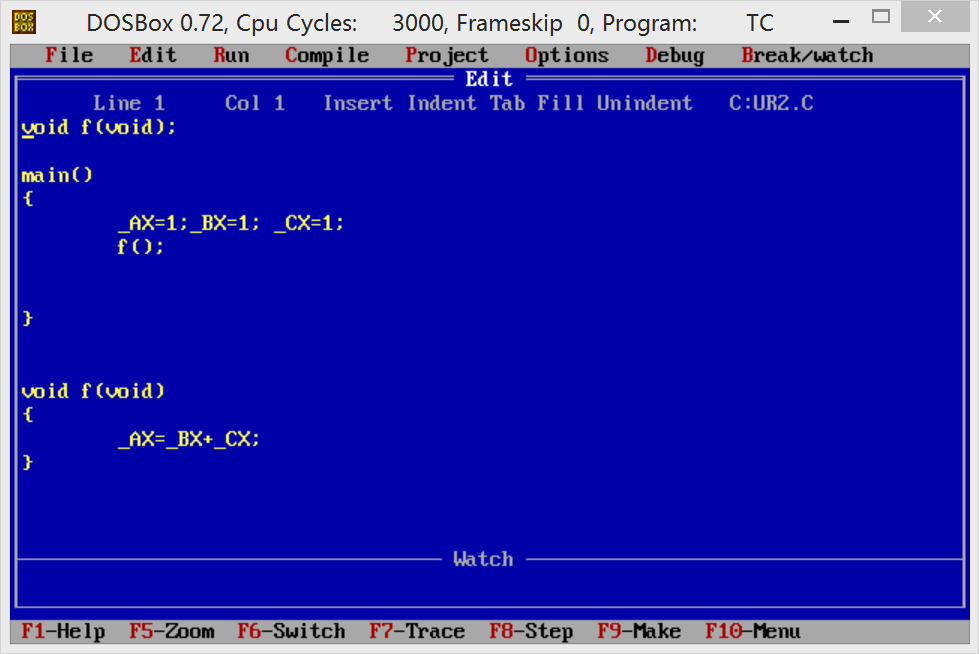
比较一下前后的地址，前：076a：0、后：076a:1fa。段地址都是cs段地址，偏移地址不同，为什么会相差 1faH呢？存疑。

0-1faH之间的代码有用吗？何用？存疑。

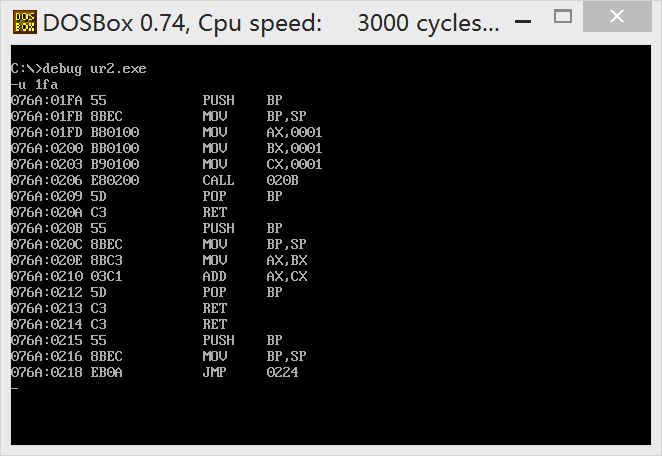
而在这里，再回头看一下在这一过程中指点迷津、至关重要的那条语句“printf(“%x\n”,main);”格式如此，是否可以理解为这里的函数名简直是在当作变量用，但是区别呢？联系呢？函数名、变量，这些区别于汇编的新生名词，在C语言反汇编为汇编语言的时候又是具体以怎样的熟悉的形式展开的呢？

相对于函数，很容易就想到了子程序，包括子程序的标号、调用、返回等都与函数感觉有某种近亲关系一样。

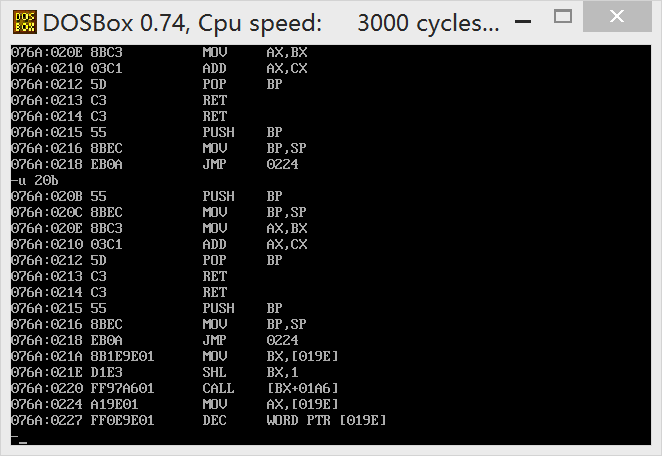
接下来的第5小点正好就是验证此项疑问的。



再用u 1fa命令：



再用u 20b命令：



至此，综合研究二课程内容完结。

1. **已思考研究并已解决问题汇总**

**1>.用C语言编程可以不用变量吗？**

可以，综合研究二得出其中一种额外选择：可以使用寄存器替换变量的使用。

**2>.函数名、变量，这些区别于汇编的新生名词，在C语言反汇编为汇编语言的时候又是具体以怎样的熟悉的形式展开的呢？**

相对于函数，很容易就想到了子程序，包括子程序的标号、调用、返回等都与函数感觉有某种近亲关系一样。变量的话，存在的形式，还不确定就是唯一的形式。

**3>.** **用debug怎么查找函数的偏移地址？**

参考语句“printf(“%x\n”,main);”，最好能举一反三。

1. **已思考研究并未解决问题汇总**

**1>.语句“printf(“%x\n”,main);”格式如此，是否可以理解为这里的函数名简直是在当作变量用，但是区别呢？联系呢？**

**2>.比较一下前后的地址，前：076a：0、后：076a:1fa。段地址都是cs段地址，偏移地址不同，为什么会相差 1faH呢？存疑。0-1faH之间的代码有用吗？何用？存疑。**

1. **研究感想（心得体会）**

不要主观臆断，不要主观臆断！本来自以为的直接”u”一下就可以的事情，结果却不是，如果没有书里一步步给的推测线索，自己可以找到这样简洁有效的方法来检测吗？如果不行，自己到底还是哪里短缺呢？是对错误的敏感不足，还是缺少一步步推敲、总结、反思的能力与意识呢？再想想，printf()用的少吗？可是它的功能自己是不是都玩转了，printf()用起来简单，难道printf()本身真的很简单吗？那要看综合研究五了。