# 一、实验目的与要求

（1）掌握处理器仿真工具LC-3软件的安装和使用方法。

（2）学会在LC-3仿真环境下编辑程序和转换成可执行目标程序的方法 。

（2）学会在LC-3仿真环境下运行和调试程序的方法 。

# 二、实验内容与方法

利用提供的安装软件包和软件使用说明文档，完成以下试验内容：

（1）安装LC-3仿真器

（2）利用LC3EDIT输入机器代码程序（0/1模式）并创建可执行目标程序。

（3）利用LC3EDIT输入机器代码程序（hex模式）并创建可执行目标程序。

（4）利用LC3EDIT输入汇编代码程序并创建可执行目标程序。

（5）利用仿真器运用对应目标程序。

（6）学习和掌握断点，单步执行等调试方法和手段。

# 三、实验步骤与过程

（依照实验内容，逐条撰写实验过程与实验所得结果：包括程序总体设计，核心数据结构及算法流程，调试过程。请附上核心代码，及注意格式排版的美观。实验提交时，以上为评分依据，请不删除本行）

**Example1：调试这个不适用乘法指令的乘法程序**

首先需要在 LC3Edit 中输入以下程序：

0011 0010 0000 0000 ;程序起始地址: x3200

0101 010 010 1 00000 ; R2 复位

0001 010 010 0 00 100 ;R4中值与R2相加 结果放置与R2中

0001 101 101 1 11111 ;R5中值减去

1 0000 011 111111101 ;如果结果>=0 转移至x3201

1111 0000 00100101 ;停止

如图1-1所示：

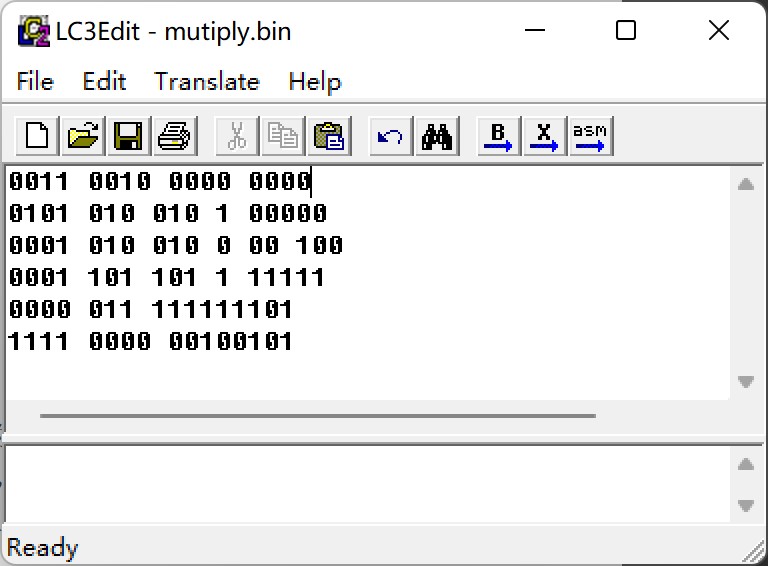


图1-1

保存为mutiply.bin ，单击 转化为.obj 文件，如图1-2所示：

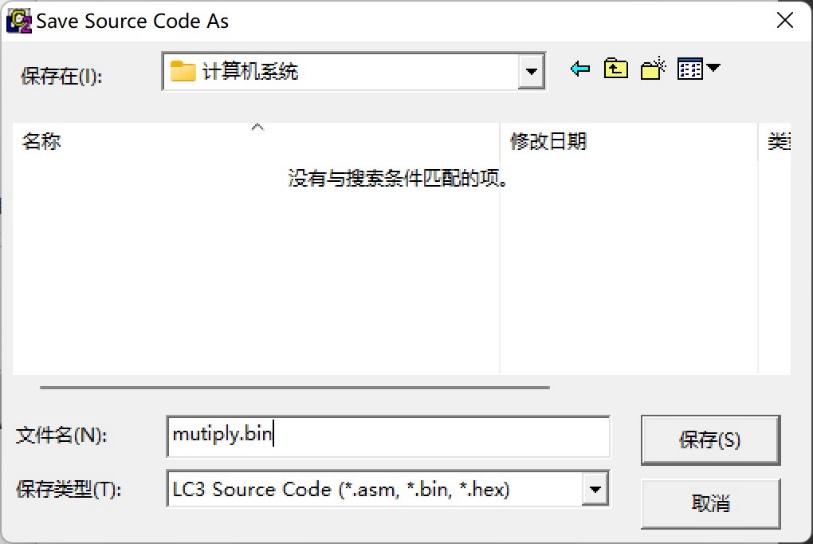


图1-2

开启模拟器，加载程序multiply.obj,如图1-3所示：

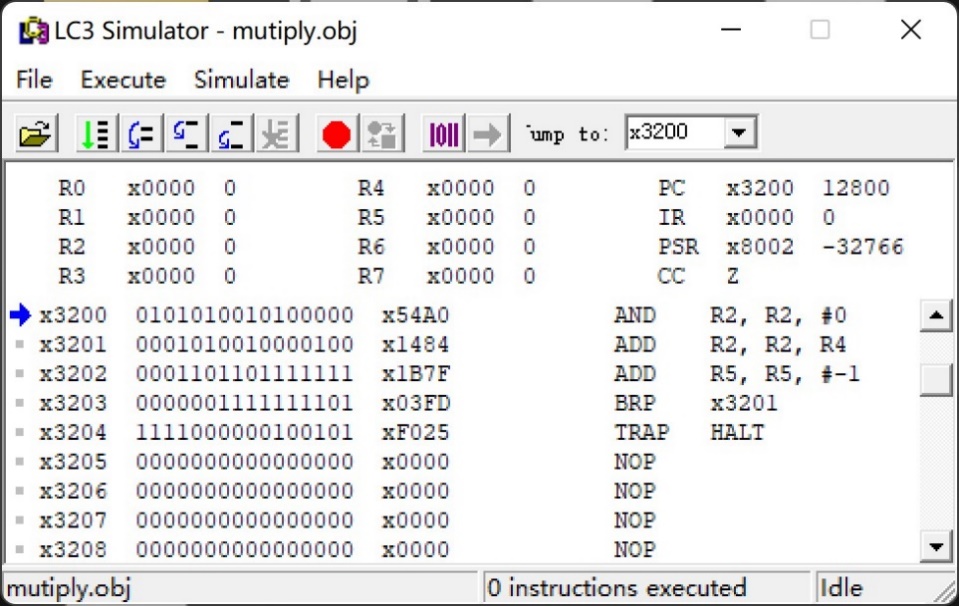


图1-3

halt 指令处设置断点，在行 x3204 处设置断点，双击最前面的灰色方框，如图1-4所示：

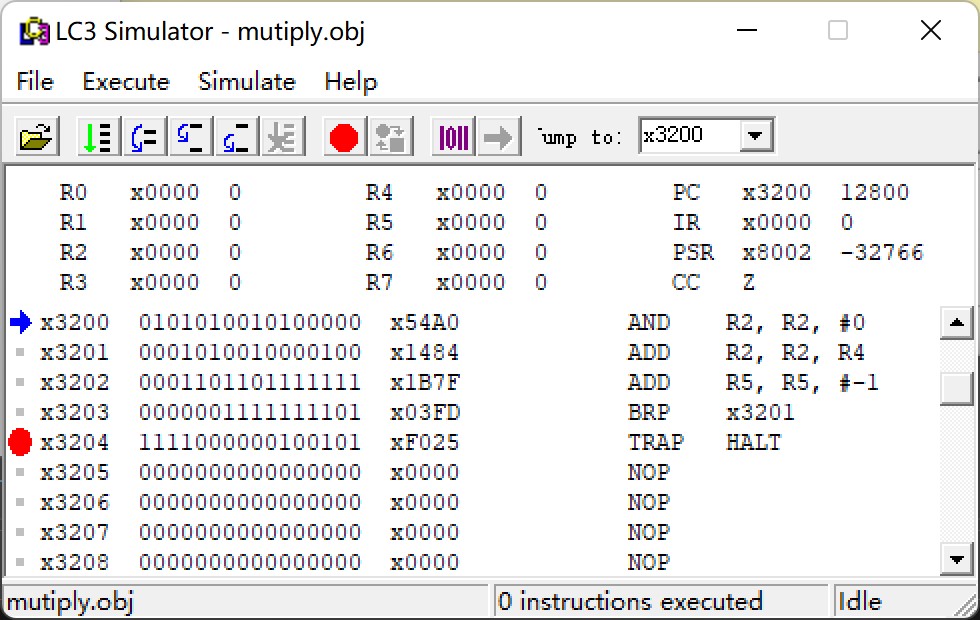


图1-4

为 R4、R5 设置相应的值，弹出 Set Value 窗口，在 Location 字段中选择 R4，在 Value 处输入“x5”，点击 Apply，然后选择 R5，输入 x3，点击 OK，如图1-5所示：

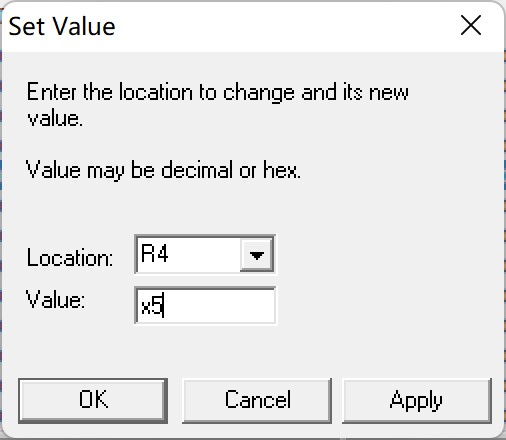


图1-5

运行程序，弹出窗口如图1-6所示：

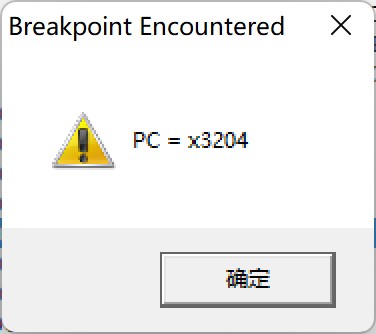


图1-6

查看 R2，应该包含最后的结果十进制 3\*5=15，但是 R2 中包含十进制 20（十六进制 x14)，如图1-7所示：

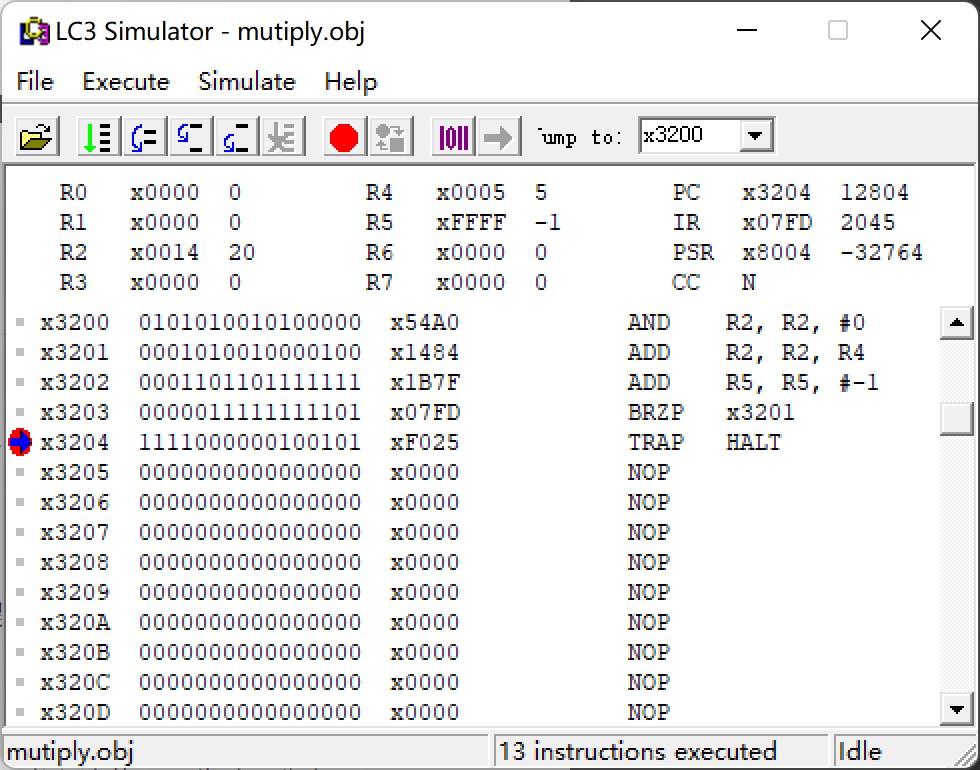


图1-7

调试乘法程序，双击 R5，然后在弹出的窗口中设置 R5 为 x3，然后点击 OK。然后点击内存区域 x3200 处，接着设置 PC 值为 x3200。接下来调试，让PC 指向下一条指令 X3201，IR 中内容为第一条指令，X54A0，R2被清零，如图1-8所示：

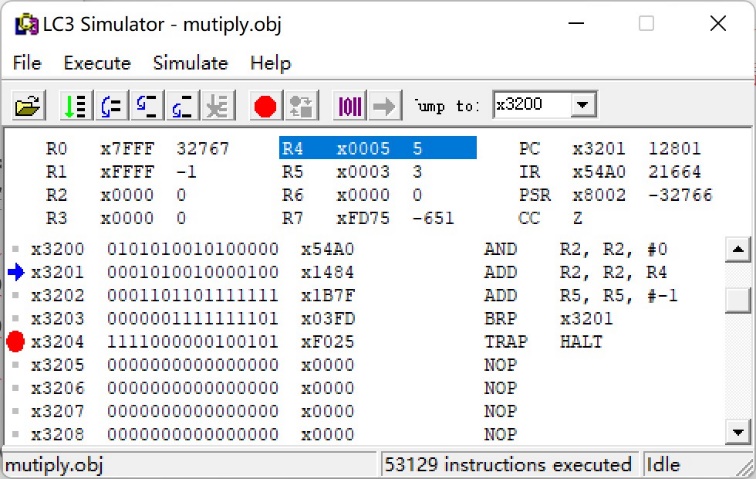


图1-8

接着调试发现每条指令都没有问题，那么问题有可能出现在循环设置的地方。

在分支指令处设置断点。在Location 处点击下拉箭头，设置 PC 为 x3203，然后点击 Add，点击 ok，设置 PC 为 x3200，R5 为 x3，如图1-9所示：

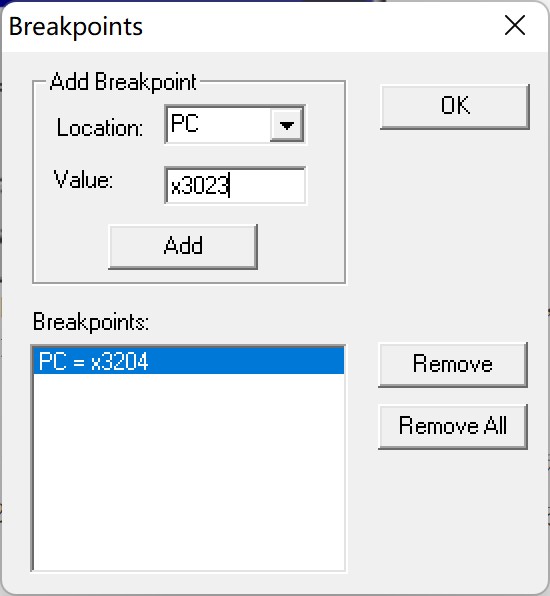


图1-9

运行程序，接着就会弹出窗口如图1-10所示：

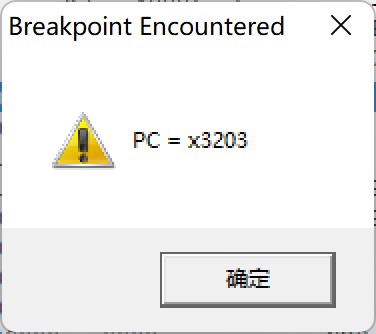


图1-10

继续调试发现当R5为0，R2为十进制15时，此时应该停止，但是状态码 Z=1，分支指令将继续执行，多做一次，这里有问题。

用 LC3Edit 修改分支指令如下： 0000 001 111111101 ;跳转到 location x3201

重新运行程序可以看到十进制15出现在R2中，如图1-11所示：

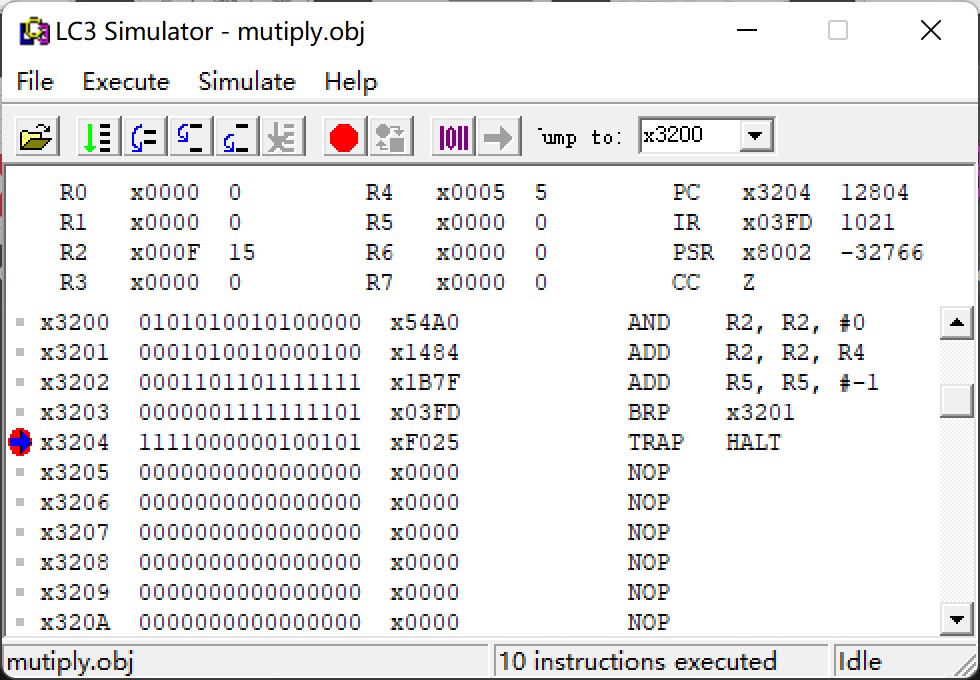
****

图1-11

**Example 2：调试程序使其输入输入并求和**

在LC3Edit中输入以下程序：

.ORIG x3000

TRAP x23 ;the trap instruction which is also known as "IN"

ADD R1,R0,x0 ;move the first integer to register 1

TRAP x23 ;another "IN"

ADD R2,R0,R1 ;两个整数相加

LEA R0,MESG ;载入字符串的地址

TRAP x22 ;输出字符串

ADD R0,R2,x0 ;sum 保存到 R0 中，并准备输出

TRAP x21 ;显示结果

HALT

MESG .STRINGZ"The sum of those two numbers is"

.END

将程序保存为asm文件如图2-1所示：

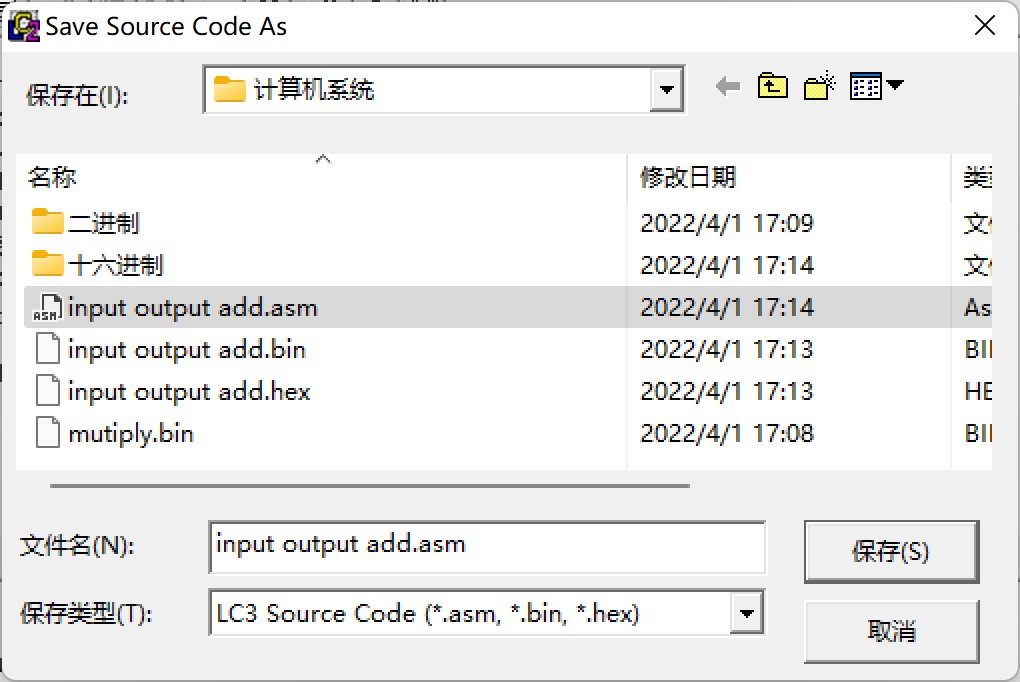


图2-1

并进行编译，如图2-2所示：

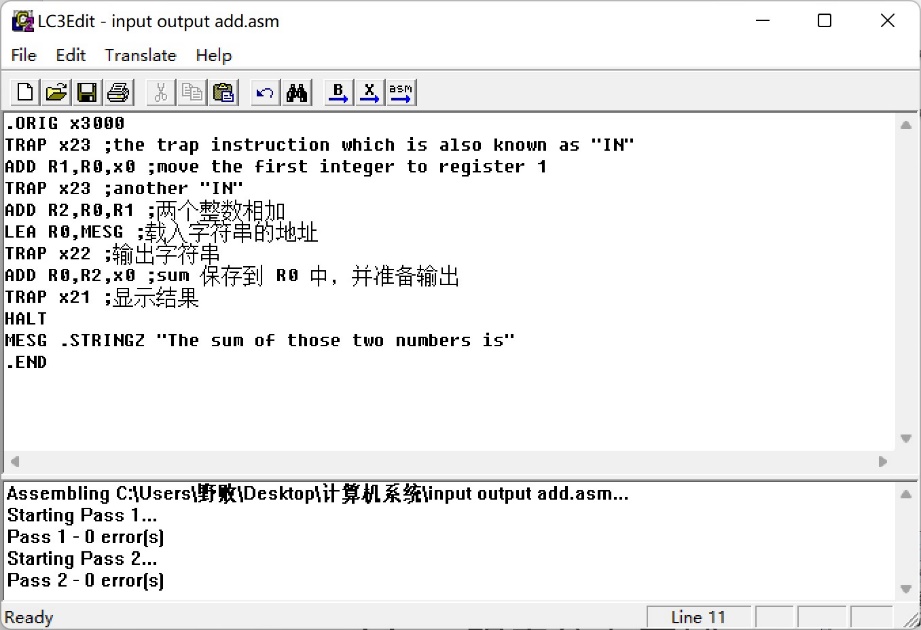


图2-2

打开 simulator，载入程序，双击 x3008 行前的小灰色方块来设置断点，如图2-3所示：

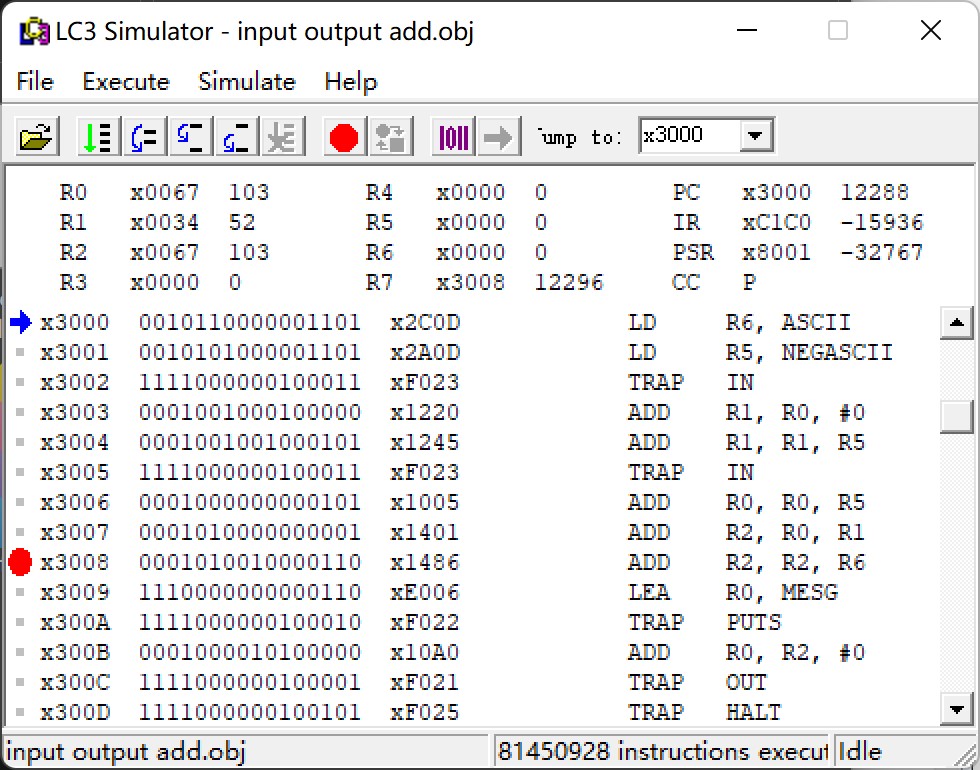


图2-3

运行程序，输入“4”，输入“3”，输出结果如图2-4所示：

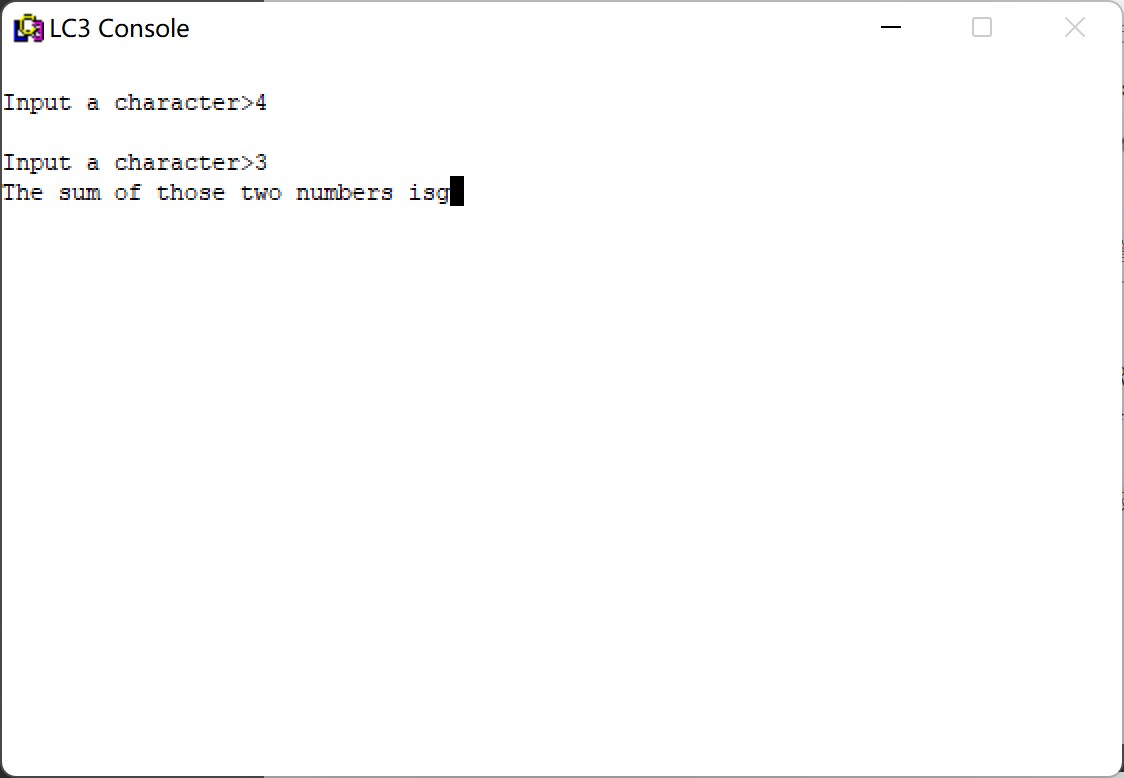


图2-4

调试程序，当输入“4”时，R0的值是 x34，当输入“3”时，显示的是 x33，相加结果是 x67。x67 代表的是“g”。

将ASCII .FILL x30 ;mask：转换成 ASCII MEGASCII .FILL xFFD0 ；mask：-x30

载入两个 mask，一个是将-x30 加到第一个数中，一个是对另一 个数做同样的操作，最后一个是在输出前将结果加上 x30。新程序如图2-5所示：

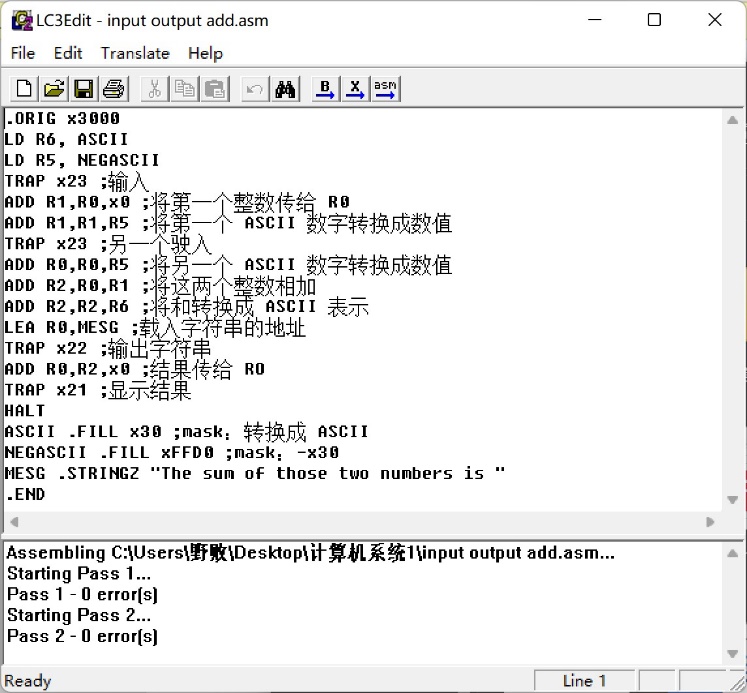
****

图2-5

由于程序改变，我们需要设置新的断点为x3010，如图2-6所示：

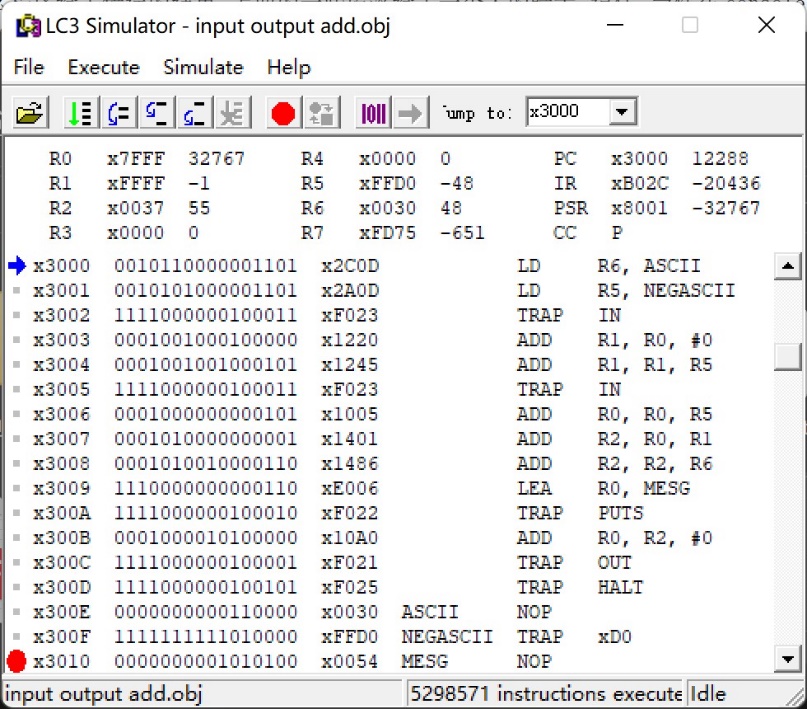
****

图2-6

运行程序就可以得到想要的结果，如图2-7所示：

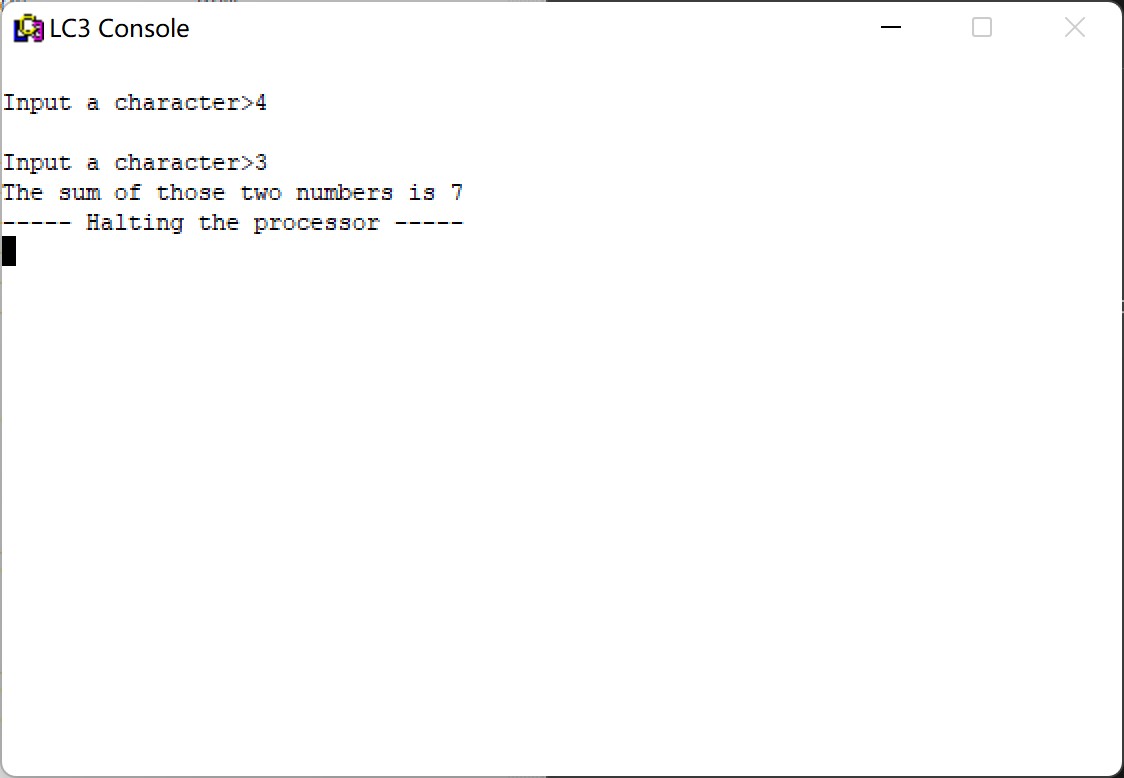


图2-7

# 四、实验结论或体会

（撰写实验收获及思考）

在这次实验中，我利用所提供的LC-3使用指南初步掌握了处理器仿真工具LC-3软件的安装和使用方法，熟悉了LC-3仿真器的基本操作，包括使用LC3EDIT输入机器代码程序（0/1模式）并创建可执行目标程序；使用LC3EDIT输入汇编代码程序并创建可执行目标程序；以及使用LC-3Simulator仿真器运行对于目标程序。

实验过程中，在LC3模拟器中运行程序以及调试程序，掌握了设置断点、单步执行等调试方法和手段。

在运行汇编程序的过程中，在正确修改代码之后发现无法输出运行结果，但寄存器内所存储的值是正确的，经过几番试验发现是设置的断点问题，修改后的代码相比原来的要长，设置的断点应该延后，重新设置断点后就可以得到预期输出结果。

希望以后能加强对汇编语言的学习。

|  |
| --- |
| 指导教师批阅意见：  成绩评定：  指导教师签字：  年 月 日 |
| 备注： |

注：1、报告内的项目或内容设置，可根据实际情况加以调整和补充。

2、教师批改学生实验报告时间应在学生提交实验报告时间后10日内。