# 实验四 汇编分支结构仿真操作

一、实验目的

熟悉**Keil**环境，通过在**Keil**环境下调试汇编分支结构程序，掌握汇编语言程序的调试方法，加深对汇编分支、循环、寄存器、数据指针、汇编语言指令、机器码等基本概念的理解，为后续程序编制和调试打下基础。

二、 实验内容

在内部RAM的以20H单元为起始地址的18个带符号数，编写程序，

1、将18个单元初始化（正负零都必须有），

2、将正数、负数及零的数目存在放在60H、61H及62H单元中，

3、将正数的最大值放入63H中，将负数最小值放入64H中。

三、实验结果

1、首先通过循环向20H单元为起始地址处加入18个带符号数。实验过程中编写程序如下所示。

MAIN: MOV A,#-08H ;设置初始值

MOV R0,#1EH ;下一地址为20

MOV R2,#13H ;18个数值

// 初始化数据

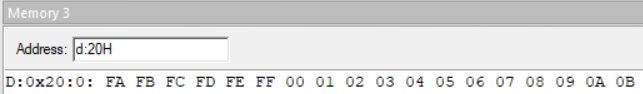
INIT: INC A

INC R0

MOV @R0,A

DJNZ R2,INIT

该程序首先设置起始地址20H和初始值-09H，然后每次循环地址加1，填入数值加1.编写完程序后进行调试，在keil中观察相应内存的值，如下图所示：



观察该结果，发现从地址20H开始，依次存入数据FA、FB、…、0A、0B一共十八个数字，其中前面一部分是负数，中间为0，后面一部分为正数，满足实验1的要求。

2&3、实验2、3要求实现统计上面18个有符号数的正数个数、负数个数、零个数、正数最大值以及负数最小值并存入相应地址中。经过理论分析可以快速得出上述结果，即为：0B、06、01、0B、FA。编写程序如下所示：

MOV 63H,#0H ;正数最大值

MOV 64H,#-1H ; 负数最小值

SJMP START

START: MOV A,@R0

DJNZ R2,NEXT

SJMP DONE

NEXT: JZ ZERO ;X=0

JB ACC.7, NEGATIVE ;X<0

SJMP POSITIVE ;X>0

ZERO: MOV A, 62H ;X=0

INC A

MOV 62H, A

INC R0

SJMP START

NEGATIVE: MOV B,64H

MOV 64H,A ;X<0

MOV A, 61H ;计算数量

INC A

MOV 61H, A

INC R0

MOV A,64H ;MIN

CJNE A,B,CHANGE1

SJMP START

CHANGE1: JNC SMALL ;负数最小值

MOV 64H,A

SJMP START

SMALL: MOV 64H,B

SJMP START

POSITIVE: MOV B,63H

MOV 63H,A

MOV A, 60H ;X>0

INC A

MOV 60H, A

INC R0

MOV A,63H ;MAX

CJNE A,B,CHANGE2

SJMP START

CHANGE2: JNC BIGGER

MOV 63H,B

SJMP START

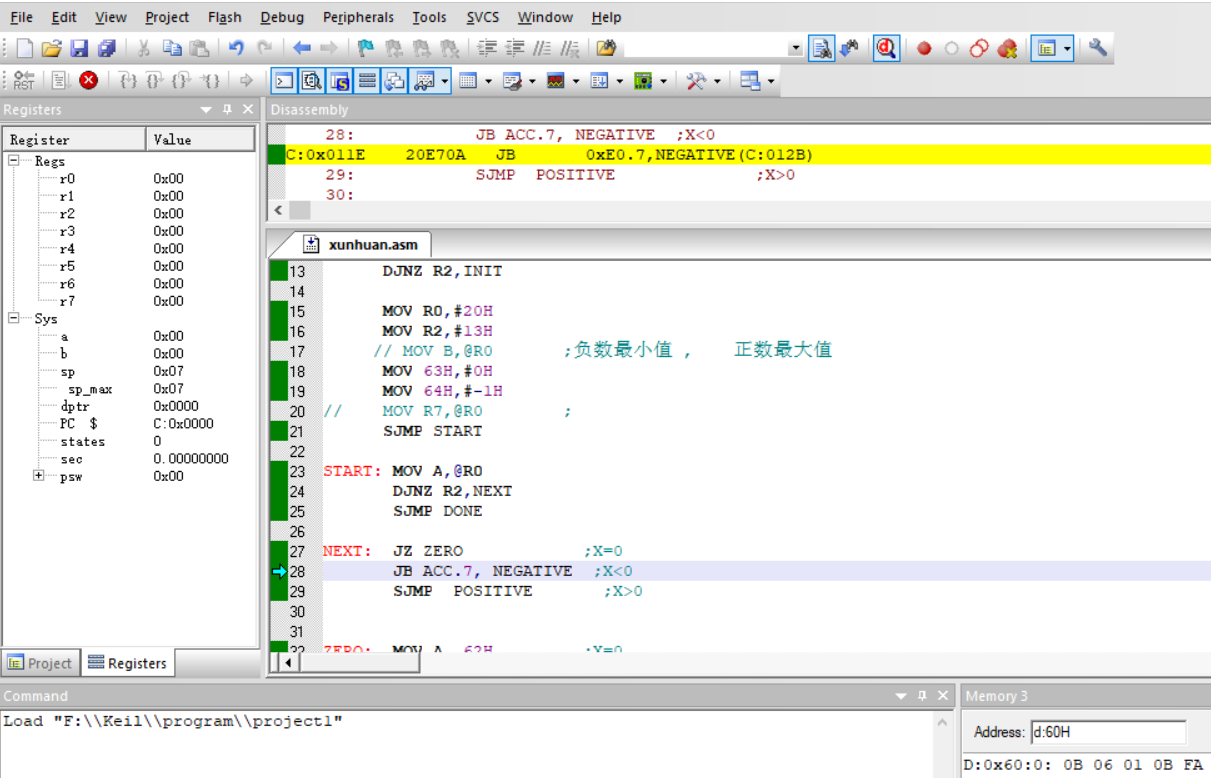
BIGGER: MOV 63H,A

SJMP START

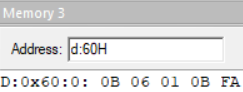
DONE: SJMP DONE

END

编写完上述程序，在Keil中进行调试，调试过程如下图所示：



经过调试，观察输出结果，即相应内存地址中是否得到结果，地址栏输入d:60H，相应内存值如下所示：



由上图可知，得到结果与理论结果相同，仿真结果满足实验2&3要求。

四、实验总结

通过本次仿真实验加深了自己对于51系列单片机指令系统的理解，尤其是对控制转移类指令、算数运算类指令的掌握。同时进一步熟悉Keil的使用方法，能够熟练的掌握工作寄存器状态的观察方法和相应内存信息的获取，对调试工作更加熟悉，过程中遇到的相关问题也加强了自己排错纠错的能力！

在实验中因为最大值最小值问题一直遇到问题导致进展较慢，后来经过不断调试和理论分析发现是因为在内存64H、63H位置处没有设置初始最大最小值，导致结果出现错误，同时发现JNC指令并不能判断正数和负数大小，只能判断相同符号大小，总之，通过本次仿真实验自己受益良多，取得相对不错的进步。