**Lab02**

**梁峻滔 PB19051175**

**1. 算法思想**

给定两个正整数，常用的求最大公约数的方法有辗转相除法和更相减损术，但是由于除法在LC-3中相对不容易实现，而减法则可以直接使用NOT和ADD指令来实现，所以采用更相减损术。算法如下：

任意给定两个正整数，以较大的数减较小的数，接着把所得的差与较小的数比较，并以两者中的大数减小数。继续这个操作，直到所得的减数和差相等为止。此时的减数或者差就是所求得的最大公约数。

例. 用更相减损术求98与63的最大公约数。

解：把98和63以大数减小数，并辗转相减：

98-63=35

63-35=28

35-28=7

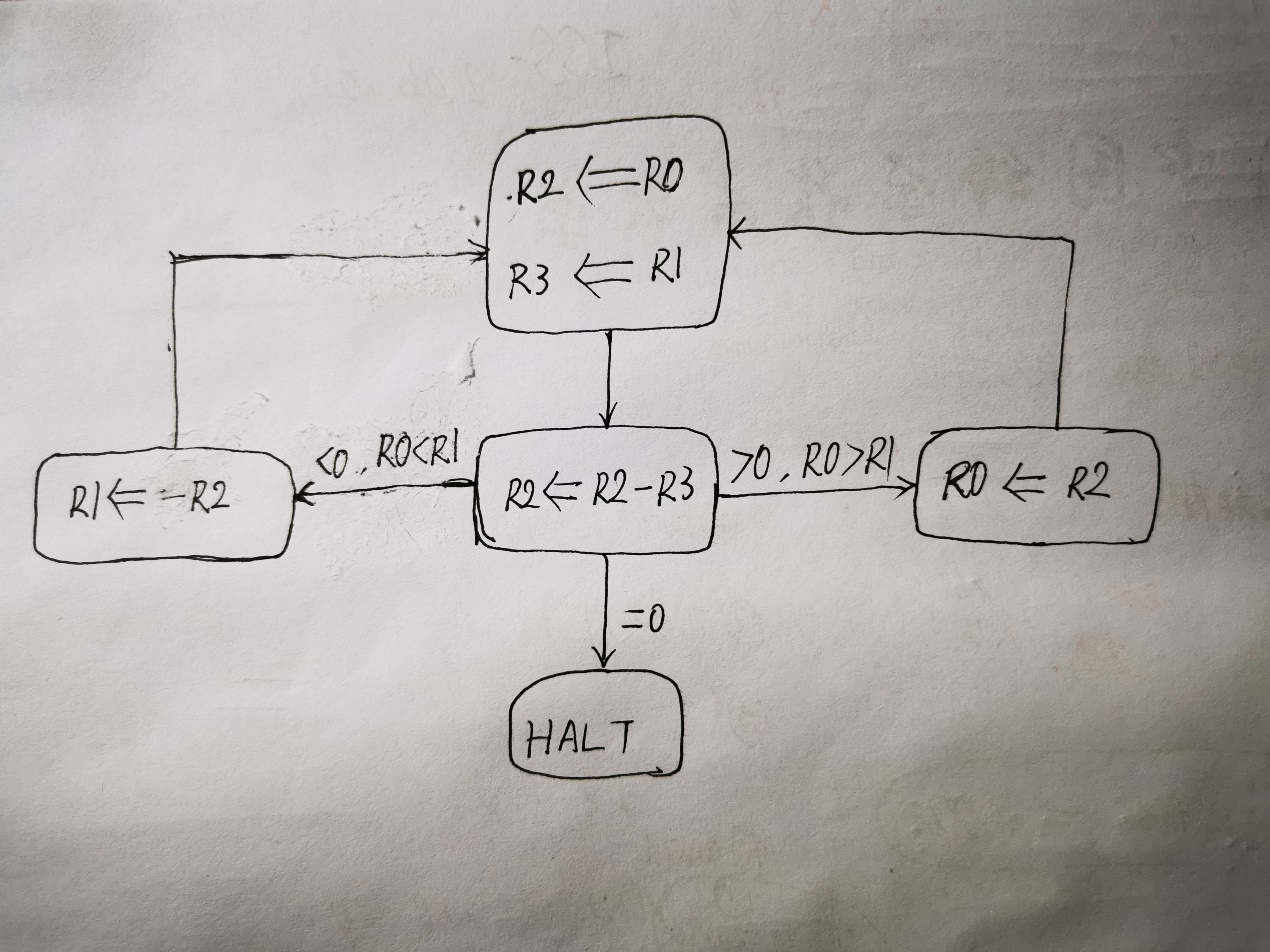
28-7=21

21-7=14

14-7=7

所以，98和63的最大公约数等于7。

在LC-3中，可以使用另外两个寄存器R2和R3来进行相减和比较，并根据相减的结果(可以同时得到差和两个操作数的相对大小)来更新R0和R1，按算法重复操作，直到相减结果为0，此时R0和R1中都将存放着最大公约数。流程图如下：

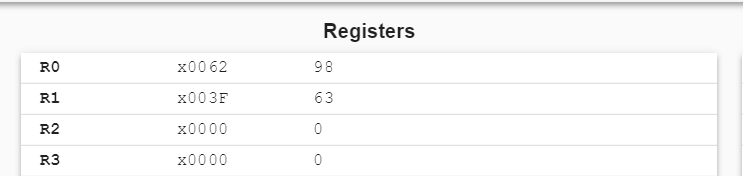


**2. 代码实现**

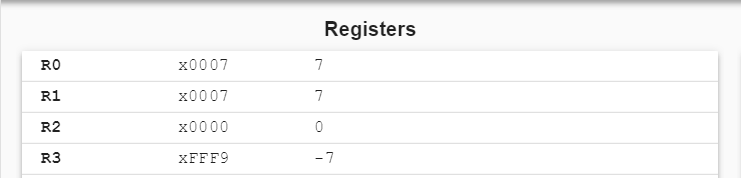
****

**3. 测试**

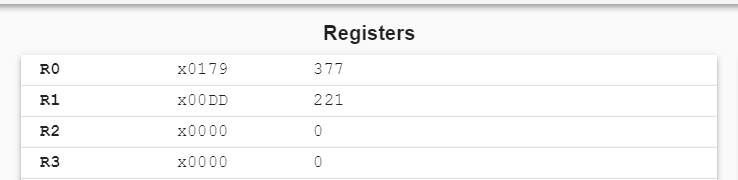
首先设置R0、R1的初始值：



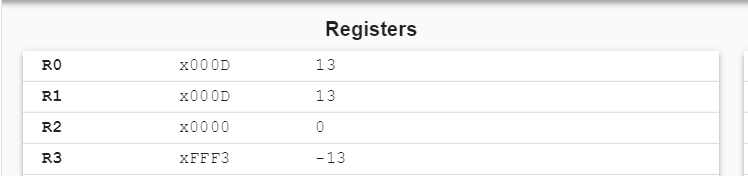
运行后得到了同所举例子相同的结果：



设置，，则预期运行后的结果为。



运行后



得到了预期的结果，程序正确地完成了任务。