Lab1: F-word

Description

As far as we can see, one word is 16 bits in LC-3. We may call a word F-word if it contains 4continuous 1.

Requierments

- Write program with LC-3 machine code(0's and 1's)
- Start your program at x3000
- The word to be checked will be stored at x3100 before running your program
- Set R2 to your answer: 1 for a F-word, 0 otherwise. Any other value will fail the tests.
- Remember to halt your program in the end

Ideas

We can use a mask like `0F00` to `and` with the word,getting the bits(for example, bit[11： 8]).And then examine the result whether equals to the mask or not.

Code

the Assembly Language version

```
        .ORIG X3000
        LD R0,#255      ;将数据加载到R0
        AND R1,R1,#0    ;初始化R1，用于存放掩码
        AND R2,R2,#0    ;初始化R2，用于存放结果
        AND R3,R3,#0    ;初始化R3，用于存放掩码与数据的与运算结果
        AND R4,R4,#0    ;初始化R4，用于存放掩码的相反数的补码
        AND R5,R5,#0    ;初始化R5，用于计数器

        ADD R1,R1,#15
        ADD R5,R5,#-13  ;计数器初始值

        AND R3,R0,R1    ;获得掩码与数据的与运算结果
        NOT R4,R1
        ADD R4,R4,#1    ;获得掩码的反码的补码
        ADD R3,R3,R4    ;计算结果

JUDGE   BRZ RESULT;判断结果是否为0
        ADD R5,R5,#1    ;R5自增
        BRZ FINAL;判断R5是否为0
        ADD R1,R1,R1    ;掩码前移

        AND R3,R0,R1    ;获得掩码与数据的与运算结果
        NOT R4,R1
        ADD R4,R4,#1    ;获得掩码的反码的补码
        ADD R3,R3,R4    ;计算结果
        BRNZP JUDGE     ;返回判断语句

RESULT  ADD R2,R2,#1
FINAL   HALT
        .END
```

the Mechine Code version

```
0011 0000 0000 0000
0010 0000 1111 1111
0101 0010 0110 0000
0101 0100 1010 0000
0101 0110 1110 0000
0101 1001 0010 0000
0101 1011 0110 0000
0001 0010 0110 1111
0001 1011 0111 0011
0101 0110 0000 0001
1001 1000 0111 1111
0001 1001 0010 0001
0001 0110 1100 0100
0000 0100 0000 1000
0001 1011 0110 0001
0000 0100 0000 0111
0001 0010 0100 0001
0101 0110 0000 0001
1001 1000 0111 1111
0001 1001 0010 0001
0001 0110 1100 0100
0000 1111 1111 0111
0001 0100 1010 0001
1111 0000 0010 0101
```

Thoughts

The first time to write a program in machine code, there are still many things to learn for me.