

Lab 1 Report

1 Introduction

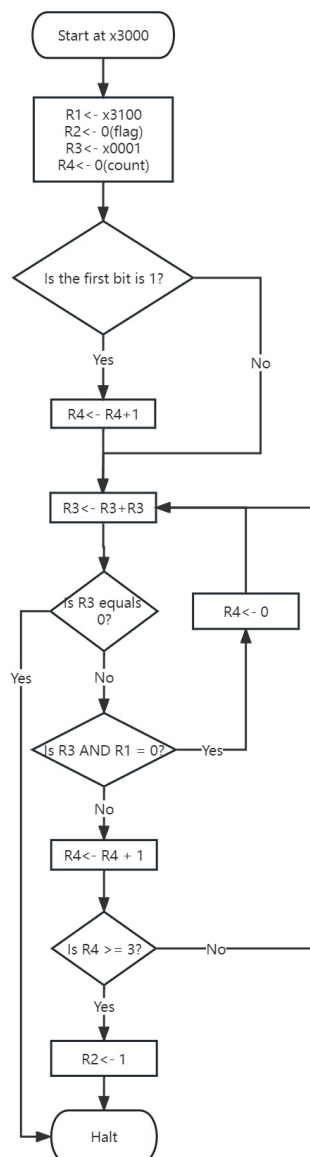
In this lab, we will write a program with LC-3 machine code to judge whether a given 16-bit value contains three consecutive 1's.

2 Solution

2.1 Algorithm

To judge whether it contains three consecutive 1's, we can use iterators. For a loop, we could check whether one of the 16 bits is 1. If so, we can add 1 to R4, which is used as a counter. If the counter equals 3, this value meets the conditions and we jump out the iteration, set R2 to 1 and halt the program. For every loop, we move the 1 in the value of R3 one bit left. Hence we can traverse every bit of the value of x3100 until we find three consecutive 1's or the traverse is done and there is no three consecutive 1's.

The flowchart of the algorithm is as followed.



2.2 Code

The code is as followed.

```
1 0011 0000 0000 0000; ORIG x3000
2
3 0010 001 011111111; load R1 with x3100
4 0101 010 010 1 00000; clear R2, to be used as a flag
5 0101 011 011 1 00000; clear R3, to be used for interation
6 0001 011 011 1 00001; set the initial value 1 of R3
7 0101 100 100 1 00000; clear R4, to be uesed as a counter
8
9 0101 000 011 0 00001; check whether the first bit is 1
10 0000 010 000000011; if the first bit is 0, the counter doesn't add 1
11 0001 100 100 1 00001; if the first bit is 1, the counter adds 1
12 0000 111 000000001; start the interation with the counter equals 1
13
14 0101 100 100 1 00000; clear R4, to be uesed as a counter
15
16 0001 011 011 0 00011; shift the 1 in R3 one bit to the left
17 0000 010 000000110; check if the interation is done
18 0101 000 011 0 00001; examinate every bit of x3100
19 0000 010 111111011; if the bit is 0, jump to line 14
20 0001 100 100 1 00001; count the number of consecutive 1's
21 0001 000 100 1 11101; the value of R4 minus 3
22 0000 011 000000010; check if there are three consecutive 1's and set R2 to 1
23 0000 111 111111000; continue the interation, jump to line 16
24
25 1111 0000 0010 0101; halt
26
27 0001 010 010 1 00001; set the flat to 1
28 0000 111 111111101; jump to line 28
```

2.3 Q&A