

# ICS-lab1 实验报告

孔浩宇 PB20000113

2022 年 11 月 9 日

## 目录

<b>1</b>	<b>实验目的</b>	<b>2</b>
<b>2</b>	<b>实验原理</b>	<b>2</b>
2.1	判断第 $i$ 位是否为 1	2
2.2	实现从第 0 位到第 $B - 1$ 位的循环判断	2
2.3	实现对为 1 的位数的记录	2
2.4	结束条件的判断	2
2.5	实现 $R_j - B$	2
<b>3</b>	<b>实验步骤</b>	<b>3</b>
3.1	初始化	3
3.2	循环	3
3.3	储存结果	3
3.4	结束	4
3.5	代码	4
<b>4</b>	<b>实验结果</b>	<b>4</b>

## 1 实验目的

对于给定的数  $A$ ,  $B$ , 利用  $LC-3$  实现对  $A$  的后  $B$  位中 1 的数量的计数。

## 2 实验原理

### 2.1 判断第 $i$ 位是否为 1

取  $E_i = 1 \times 2^i$ , 则除第  $i$  位为 1 外,  $E_i$  的其他位均为 0. 则

$$E_i \& A[15:0] = E_i \cdot A_i.$$

即对  $A$  的后  $B$  位依次与对应的  $E_i$  相与, 若结果为 0 说明该位为 0, 否则为 1.

### 2.2 实现从第 0 位到第 $B-1$ 位的循环判断

对于  $E_i$ , 利用  $ADD$  指令, 显然  $E_{i+1} = E_i + E_i$ , 不妨设  $E_i$  存储在寄存器  $R_k$  中, 则

$$ADD\ R_k, R_k, R_k \Rightarrow \text{此时 } R_k \text{ 中的数为 } 2E_i = E_{i+1}.$$

在  $B$  次循环中, 结束时对  $E_i$  进行以上操作, 可得到下一位判断所需  $E_{i+1}$ .

### 2.3 实现对为 1 的位数的记录

使用寄存器  $R_j$ , 初始化为 0, 循环判断过程中, 如果该位为 1 (即  $E_i \& A \neq 0$ ), 则  $R_j + 1$ , 否则跳过对  $R_j$  的操作.

### 2.4 结束条件的判断

使用寄存器  $R_l$ , 初始化为 0, 每次循环判断过程中,  $R_l + 1$ , 则当每次循环末尾, 若  $R_l - B < 0$ , 继续循环; 若  $R_l - B = 0$ , 说明已经判断了后  $B$  位, 停止循环.

### 2.5 实现 $R_j - B$

由于  $LC-3$  指令集中没有直接进行减法的指令, 需要自己实现. 首先在  $R_b$  中存储  $B$

$$R_j - B = R_j + (NOT(B) + 1) \Rightarrow \text{对 } R_b \text{ 进行 } NOT \text{ 指令后再 } +1, \text{ 则 } R_j + R_b = R_j - B.$$

### 3 实验步骤

#### 3.1 初始化

(1) 读入  $A, B$

将存  $A$  的地址  $x3100$  写入地址  $x3012$  的内存, 利用寄存器  $R_0$  来保存  $A$

1010 000 0 0001 0001 ;  $R_0 = M[M[x3012]] = M[x3100] = A$

将存  $B$  的地址  $x3101$  写入地址  $x3013$  的内存, 利用寄存器  $R_1$  来保存  $B$ ,

1010 001 0 0001 0001 ;  $R_1 = M[M[x3013]] = M[x3101] = B$

并将  $B$  取反加 1

1001 001 001 000 000 ;  $R_1 = NOT R_1$

0001 001 001 1 00001 ;  $R_1 = R_1 + 1, now, ADD R_1 = sub B$

(2) 初始化其他变量

0101 010 010 1 00000 ;  $R_2 = 0, the result$

0101 011 011 1 00000 ;  $R_3 = 0$

0001 100 011 1 00001 ;  $R_4 = 1 (E_i)$

#### 3.2 循环

(1) 循环结束判断条件

0001 110 011 0 00 001 ;  $R_6 = R_3 + R_1 = R_3 - B$

0000 010 000 0 00 110 ;  $PC + 6$  if  $R_6 = 0$

(2) 循环过程

0101 101 000 0 00 100 ;  $R_5 = R_0 \& R_4$

0000 010 000 0 00 001 ;  $PC + 1$  if  $R_5 = 0$

0001 010 010 1 00001 ;  $R_2 = R_2 + 1$

0001 100 100 0 00 100 ;  $R_4 = 2 * R_4$

0001 011 011 1 00001 ;  $R_3 = R_3 + 1$

0000 111 111 111 000 ;  $PC - 8$

#### 3.3 储存结果

将存结果的地址  $x3102$  写入地址  $x3101$  的内存, 将寄存器  $R_2$  的内容写入地址  $x3102$

1011 010 0 0000 0001 ;  $M[M[x3011]] = M[x3102] = R_2$

### 3.4 结束

利用 Trap 指令结束程序

1111 0000 0010 0101 ; the end

### 3.5 代码

1	0011 0000 0000 0000	;begin at x3000
2	1010 000 0 0001 0001	;R0=M[M[x3012]]=M[x3100]=A (x3000)
3	1010 001 0 0001 0001	;R1=M[M[x3013]]=B (x3001)
4	1001 001 001 000 000	;R1=NOT R1
5	0001 001 001 1 00001	;R1=R1+1,nowtime,any num ADD R1 =sub B
6	0101 010 010 1 00000	;R2=0,the result
7	0101 011 011 1 00000	;R3=0
8	0001 100 011 1 00001	;R4=1
9	0001 110 011 0 00 001	;R6=R3+R1=R3-B
10	0000 010 000 0 00 110	;PC+6 if R6=0
11	0101 101 000 0 00 100	;R5=R0 & R4
12	0000 010 000 0 00 001	;PC+1 if R5=0
13	0001 010 010 1 00001	;R2=R2+1
14	0001 100 100 0 00 100	;R4=2*R4
15	0001 011 011 1 00001	;R3=R3+1
16	0000 111 111 111 000	;PC-8
17	1011 010 0 0000 0001	;M[M[x3011]]=M[x3102]=R2
18	1111 0000 0010 0101	;the end
19	0011 0001 0000 0010	;M[x3011]=x3102
20	0011 0001 0000 0000	;M[x3012]=x3100
21	0011 0001 0000 0001	;M[x3013]=x3001

## 4 实验结果

选择评测实验

☒ lab1 ☐ 自定义

测试样例，样例之间以逗号分割

13:3, 167:6, 32767:15, 1:0

代码文本

调试模式



如果指令数过多，无法显示日志，可以尝试减小 `单样例最大指令数` 的值

评测

机器码评测

4 / 4 个通过测试用例

- 平均指令数: 57.25
- 通过 13:3, 指令数: 33, 输出: 2
- 通过 167:6, 指令数: 56, 输出: 4
- 通过 32767:15, 指令数: 130, 输出: 15
- 通过 1:0, 指令数: 10, 输出: 0