Lab 0 Report

3160105287 Circle Lin

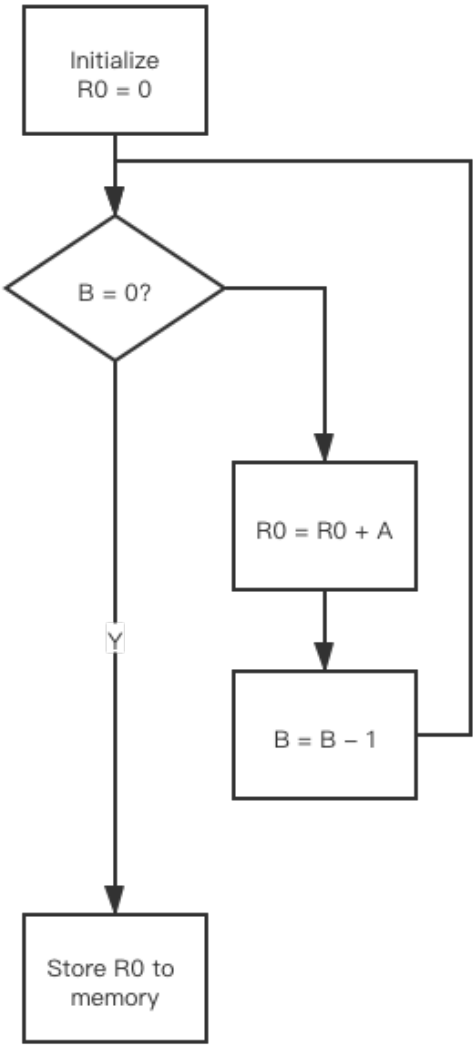
This lab requires to write a LC-3 program in machine language to perform *multiplication*. Since the LC-3 ISA does not have a multiplication instruction, we have to implement by a sequence of *additions*, that is, to add A for B times.

Algorithm

The algorithm is:

1. Get the two numbers from memory (use load instructions);
2. Do additions in a loop;
3. Store the answer back to the memory;

The loop structure is shown in the figure.



Code

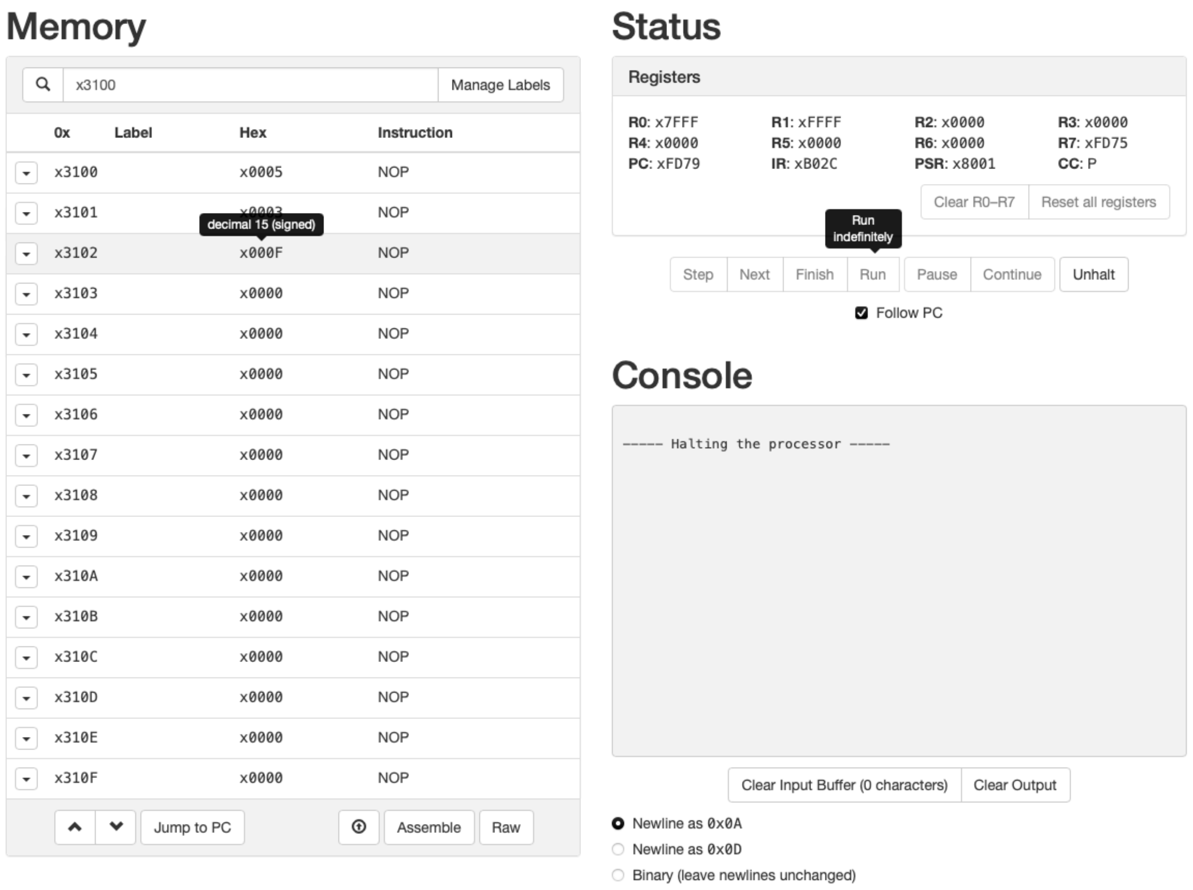
```
0011 0000 0000 0000     ; Not an instruction.
                        ; This line tells the simulator
                        ; to start the program from x3000.

0101 000 000 1 00000    ; clear R0
0010 001 011111110      ; load A from x3100 into R1
0010 010 011111110      ; load B from x3101 into R2

0000 010 000000011      ; jump to end if B is 0
0001 000 000 000 001    ; add A to answer R0
0001 010 010 1 11111    ; decrease B by 1
0000 111 111111100      ; jump back to the loop

0011 000 011111010      ; store R0 to x3102
1111 0000 0010 0101     ; halt
```

Test Results



Memory

x3100

0x	Label	Hex	Instruction
x3100		x0005	NOP
x3101		x0003	NOP
x3102		x000F	NOP
x3103		x0000	NOP
x3104		x0000	NOP
x3105		x0000	NOP
x3106		x0000	NOP
x3107		x0000	NOP
x3108		x0000	NOP
x3109		x0000	NOP
x310A		x0000	NOP
x310B		x0000	NOP
x310C		x0000	NOP
x310D		x0000	NOP
x310E		x0000	NOP
x310F		x0000	NOP

Status

Registers

R0: x7FFF **R1**: xFFFF **R2**: x0000 **R3**: x0000
R4: x0000 **R5**: x0000 **R6**: x0000 **R7**: xFD75
PC: xFD79 **IR**: xB02C **PSR**: x8001 **CC**: P

☑ Follow PC

Console

```
----- Halting the processor -----
```

● Newline as 0x0A
○ Newline as 0x0D
○ Binary (leave newlines unchanged)