Lab 3 Report

*Max Saunders*

*Msaunders3772@floridapoly.edu*

*EEL4768: Computer Architecture and Organization*

*Due Date: 20 September 2019*

1. **Project Description**

The objective of this assignment is to continue to learn basic principles of MARS MIPS and design a program that increments from 0 to 15 in both decimal and hexadecimal using a loop.

1. **Program Design**

Before the start of the main function, we set two variables for formatting purposes, newline “\n” and space which is actually a “tab”. The actual function of the program is one big while loop, that is set so that the loop will continue as long as $t1 is not greater than 15. Since $t1 is our counter it also serves to store the value that will be printed out in each increment. The program then sets $v0 to 4 to print the integer and then changes the value to 34 in order to print the value as a hexadecimal after printing a space between the two, this occurs during each incrementation. Then when $t1 has incremented to 16 the loop branches to the Exit label and exits the program.

1. **Symbol Table**

**Table 1: Registers Used**

|  |  |
| --- | --- |
| Register | Use |
| $v0 | Tells Mars what to do when syscall is called. Used to print values |
| $t1 | The first register used, used to store the counter which determines if the loop will continue, and determines the value that will be displayed |
| $a0 | The argument for the functions, used in syscall. Set to the value we wish to print, including text and newlines |

1. **Learning Coverage**

* Basic Operations in MIPS
* The usage of loops
* The usage of labels
* If-statements
* Incrementation
* Printing in hexadecimal

1. **Prototype in C-Language**

**5.1 Code in C**

#include <stdio.h>

int main() {

int hex = 0x00;

int Dec = 0;

printf("Dec: %d\t", Dec);

printf("Hex: %x\n", hex);

int i = 0;

while (i < 15)

{

i++;

Dec++;

hex++;

printf("Dec: %d\t", Dec);

printf("Hex: %x\n", hex);

}

return 0;

}

**5.2 Code in MARS MIPS**

.data

NewLine: .asciiz "\n"

Space: .asciiz " "

.text

main:

li $t1, 0

Loop: bgt $t1, 15, Exit #exit condition

move $a0, $t1

li $v0, 1 #Print value

syscall

la $a0, Space

li $v0, 4

syscall

move $a0, $t1

li $v0, 34

syscall #print in hexadeciaml

la $a0, NewLine # Print /n

li $v0, 4

syscall

addi $t1, $t1, 1 #increment

j Loop #loop

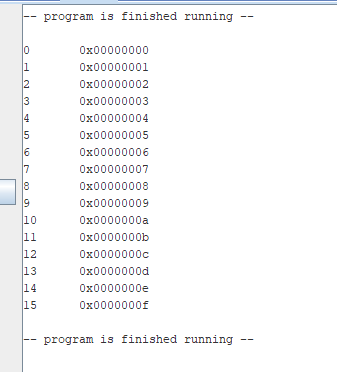
Exit: li $v0, 10 #exit

syscall

1. **Test Plan and Results**

The test plan for this program is pretty straight forward since it should run the same exact way every time it’s run. Since it ran correctly every time it was ran, I can call it a success.

**Conclusion:** The program successfully incremented from 0 to 15 and displayed the values in decimal and hexadecimal in a neat fashion, so the program performed perfectly.



1. **References**
   1. Lab Manual
      1. EEL4768-Lab3.pdf on CANVAS
   2. MARS Simulator
      1. The MARS simulator for MIPS processors