Lab #4 Lab Report

Math for Microprocessors: ALU and FPU

William Ah Tou

May 9, 2016

ECEGR 2220

ALU interface design:

I attempted to create this ALU by using the control variable to choose the supported functions. Using a case/when statement I hoped that I would be able to select each function properly. However, I ran into problems when trying to call the adder\_subtracter and shift\_register functions. After the control variable was used and the outputs were computed, the zero variable would be updated if the output computed came out to zero value.

ALU test:

Because I was not able to properly get the control variable to work, I had to “comment out” the portions of code that would not compile. This left the test to only test the “and” and “or” sections. I did, however, write code that would implement the other cases had they been working.



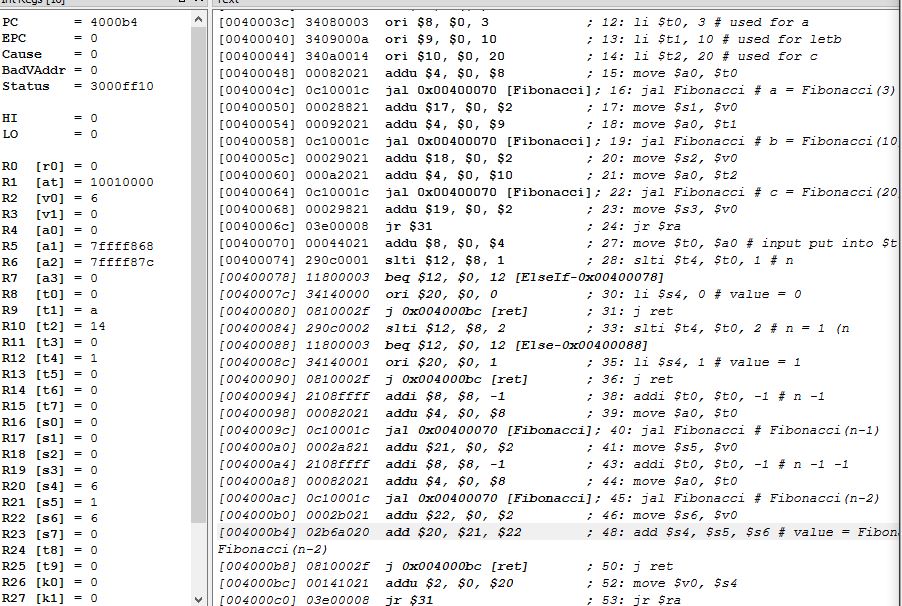
Part 2:

Question 1:

The code for the temperature in ASM created .float variables for the various numbers that were needed for conversion. The problem I ran into on this particular problem was loading the words into their float registers. Had this been successful I am confident that the rest of the code is fairly accurate in converting the user’s input into Celsius and Kelvin as nothing more that basic multiplying and adding or subtracting was needed to convert.

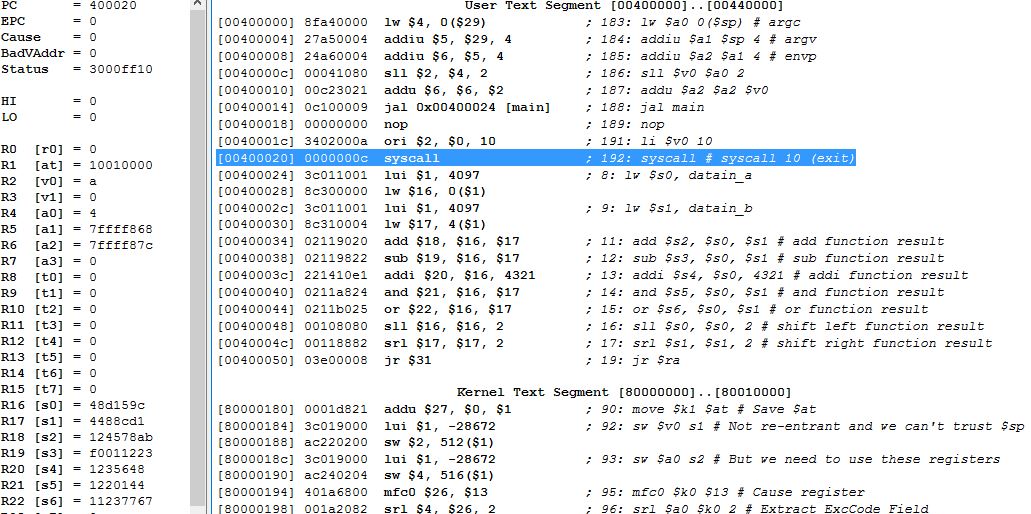
Question 2:

The Fibonacci code works until the function needs to return to the main part of the code. At this point it loops to when previously called inside itself.



Question 3:

Using the $s0-$s6 registers to hold the results of the function testing, this segment of code successfully ran. These results show what the ALU test results should be.



Fibonacci limit:

A limit to the Fibonacci sequence would eventually occur as the numbers would be beyond the scope that the registers can hold.