18-240 LAB 4 SPRING 2018

I seek knowledge in all things, but especially about the subroutine linkage behavior of the P18240. Would you please answer the following questions?

1) What is the biggest value of n for which tetrahedral(n) will still fit in a 16-bit unsigned number?

2) In the course of calculating tetrahedral(5), which subroutine calls are made? List, in order, all the subroutine calls and the parameter values used (i.e. tetrahedral(5) calls tetrahedral(4)...). Can you speed up the calculation by re-ordering the computation {the calls to tetrahedral(n-1) and triangular(n)}? How about by caching the results of previous calls?

3) Please provide a program to calculate tetrahedral(n) for an n value (that has been provided in a memory location). How many instructions are used in this program? Your program should use two subroutines:

   x = tetrahedral(n)

   x = triangular(n)

   where the n parameters and x return values are 16-bit unsigned values. Also, assume n is a positive value no larger than your answer in question 1. In other words, you need not do any error detection on input parameters. I know you are enough of a professional to verify your program's correctness with the sim240 tool.

4) How many cycles are spent in the subroutine-call "overhead" for each instance of calling tetrahedral(n). Specifically, count the number of cycles spent in passing and finding the arguments, saving and restoring registers, passing and finding the return value. Don't forget to count the number of cycles spent in the JSR and RTN instructions themselves.

5) In general, what factors affect the number of overhead cycles for a subroutine call? (Name at least 3 and explain.)

After you have spent some time in lab, you'll no doubt be able to answer the following questions that trouble me:

6) How many Logic Elements does the processor and memory occupy?

7) In simulating the SystemVerilog, how many total cycles does it take to calculate tetrahedral(3)? What percentage of these cycles are "subroutine call" overhead? Since the P18240 module dumps state to the command line for every cycle, you can redirect the output into a file (using shell redirection) for close examination.