# Homework #2

Pull the initial files for this homework into your private git repository by changing into that directory and executing the command "git pull". Create a "solution document" in the *hw02* directory (txt, doc, docx, or pdf) that includes your answers to the problems below. Then add, commit, and push the solution document to your GitHub repository, along with any additional files requested.

1. (35 points) The Gregory-Leibniz algorithm can be used to determine the value of π. Each iteration is guaranteed to get closer to the true value. Analyze the RTL simulations of the Cortex-M0 processor to find the following (5 points each):
   (a) average cycles-per-second performance for the fibonacci.c program
   (b) average cycles-per-second performance for the gregoryleibniz.c program
   (c) average Cycles-per-Instruction (CPI) for the fibonacci.c program
   (d) average Cycles-per-Instruction (CPI) for the gregoryleibniz.c program
   (e) average number of memory transactions per cycle for the fibonacci.c program
   (f) average number of memory transactions per cycle for the gregoryleibniz.c program
   (g) explain any discrepancies between the Fibonacci & Gregory-Leibniz in (a)-(f)

   Put your answers in your solution document. Be sure to increase the length of the simulation as needed to improve the accuracy of your estimation. Commit the simulations needed to determine your answer (for the gregoryleibniz.c program only, along with any scripts you wrote to determine your answer) in the *hw02/p1* directory of your *git* repository and push the commit to GitHub. Ensure that you update the *hw02/p1/sim/Makefile* if needed so that the code builds and executes properly with the *make* and *make sim* commands. Finally, add a new target to the Makefile called "answers" so that your script(s) execute properly with the *make answers* command (again, for the gregoryleibniz.c program, only).

   WARNING: Please do not commit files that are not needed to build your design, such as *elf* files and the *ram.bin* file, as well as the *work* subdirectory. If such files are committed to your repository, a few points will be deducted.

2. (45 points) Modify the DW_memctl simulation to perform 4 read operations, each of 16 bytes. The first and second read should be to the same row and bank, while the third should be to a different row (same bank), and the fourth should be to a different bank. Then for transactions 2, 3, and 4, answer the following questions:
   (5 points) How did you determine the address for the transaction?
   (5 points) What latency would you expect for this transaction?
   (5 points) What latency was simulated for this transaction? Is there a discrepancy?

   See the Project 1 section of the course web-page, for more info on the DW_memctl module.

   Commit the simulations needed to determine your answer in the *hw02/p2* directory of your *git* repository and push the commit to GitHub. Ensure that you update the

*hw02/p2/sim/Makefile* if needed so that the code builds and executes properly with the *make* and *make sim* commands.

WARNING: Please do not commit files that are not needed to build your design, such as *elf* files and the *ram.bin* file, as well as the *work* subdirectory. If such files are committed to your repository, a few points will be deducted.

3. (30 points) Find the power consumed by the Cortex-M0 processor for a gate-level simulation, running both the *Fibonacci* and *Gregory-Leibniz* programs. Plot the Dynamic (*Switching + Internal)* and Leakage power every 100 cycles and include the plot in your solution document. Make note of any difference you see in the power used by the processor while running the two different programs.

   You need only simulate for 9,500 cycles for each program. Ensure that you update the *hw02/p3/sim/Makefile* if needed so that the code builds and executes properly (for the *Gregory-Leibniz* program only) with the *make* and *make sim* commands, and the *hw02/p3/synth/Makefile* so that the power-estimation executes properly with the *make ptpx* command. If you created scripts to help find your answer, please add and commit these files and add a new target to the Makefile called "answers" so that your script(s) execute properly with the *make answers* command. Remember to push the commit to GitHub.

   WARNING: Please do not commit files that are not needed to build your design, such as *vcd* files. If such files are committed to your repository, a few points will be deducted.