Description:

The objective of this lab was the create one kernel that could do a matrix multiply on two matrices. And then do a matrix add with another matrix. The matrices A,B,C are shown below in matrices 1, 2 and 3.

Summary of the outcome:

The final results of this lab was a 2x6 matrics as shown in matrix 4. The final program successfully ran on the fpga after being compiled.

Main hurdles and difficulties:

There were three main difficulties encountered in this lab. The first was figuring out how to figure out the indices for the matrix add after doing the matrix multiply. The result ended up being the same way to calculate the indices for the result matrix.

After that I had trouble setting the kernel arguments correctly since there were two arguments to add to the kernel from the Matrix Multiply example that I used as a base, the height of width of matrix C and the global buffer that held matrix C. I had incorrectly set the indices so I was getting a CL_INVALID_KERNEL_ARGS error from the clEnqueueRangeNDKernel command.

Finally I had trouble compiling the proram for the FPGA because it was the first time I had done it. The problem was .

Things you learned from this lab:

I think one of the valuale takeaways from this lab was to make sure that your kernel arguments are set correctly because that could throw off the entire program and end up being not what you intented to happen. Also, I learned how to compile an openCL program for an FPGA which is valuable because the use of openCL is to run on hardware accelerators such as FPGAs.

Code can be found at:

https://github.com/pshiverick/lab1-matrixadd