UMass Lowell EECE.6540 Heterogeneous Computing

Lab 2: Calculate Pi (π) using OpenCL

Due Date: See the course syllabus or piazza page.

rev:10/30/18

Objectives

- Learn the basics of OpenCL programming
- Learn the basics of OpenCL kernel design
- Learn to apply data and task decomposition in designing parallel program

Description

In this lab, you will design an OpenCL program that computes Pi (π). There are many different algorithms and methods to calculate Pi. In this lab we use the following formula:

$$\mathbf{Pi/4 = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \frac{1}{11} + \frac{1}{13} - \frac{1}{15} + \frac{1}{17} - \frac{1}{19} + \frac{1}{21} - \frac{1}{23} + \ldots..}$$

You need to design your OpenCL program so that:

(1) It calculates Pi up to at least two fractional digits (i.e., 3.14…);
(2) It calculates the results using only **one** OpenCL kernel;
(3) It releases host side and device side resources before completion;
(4) It has error checking at the important steps.

In addition, this lab is to get you familiarize with the OpenCL development environment, tools and particularly the design flow on an FPGA based platform. You will practice the commands, and perform the compilation and execution steps in a Linux environment. While the lab can be completed on a Mac OS platform where OpenCL is natively supported, we expect you to eventually use the Linux server we provide, to which you need to login remotely over VPN (login instructions have been given on Blackboard).

Helpful Notes

You may consider a similar design methodology as the MapReduce example. Consider partitioning the computation of (e.g., `x-y`) to work items. However, one of the challenges is that "atomic_add" and "atomic_inc" do not have their floating-point counterparts. As a result, you would need to accumulate the intermediate sum within a workgroup using a different method. For example, you can use an array of floating point numbers in local memory to store the intermediate results. The local reduction is thus somewhat different. Host side program can be extended to do global reduction.

Deadline

See course syllabus and latest announcement on Blackboard.

Deliverables

A Lab report that contains the following sections:

1. Description of the lab in your own words
2. Summary of the outcome (final results, working, partial working, etc.)
3. Main hurdles and difficulties (expected to include some specifics)
4. Things learned from this lab (valuable takeaways)
5. Suggestions (Optional)
6. Link to your final source code on github

Reference

[1] Lab Assignment materials posted on git repository :
https://github.com/ACANETS/eece-6540-labs