Exercise Manual

for

Introduction to OpenCL™ for Intel® FPGAs

Lab Exercise 3

Software Requirements

CentOS 7 Linux* OS
Eclipse IDE
Intel® FPGA SDK for OpenCL™ version 19.1
Intel® Quartus® Prime Pro software version 19.1 with Arria® 10 family
Intel® Code Builder for OpenCL™ (Needed to run fast emulator)

Exercise 3

NDRange Kernel

In this exercise, we will convert kernel that you created in exercise 2 into an NDRange kernel.

Step 1. Convert and Compile the Kernel

____ 1. Reopen the **SimpleOpenCL** project in Eclipse if it is not already open, and change to the C/C++ Eclipse perspective.

____ 2. Open **SimpleKernel.cl** by using **File -> Open File…**

____ 3. Save it as **SimpleKernel_For.cl**

We will come back to the for loop version of the kernel in the next exercise.

____ 4. Reopen **SimpleKernel.cl**

____ 5. Convert the kernel into a ND Range Kernel.

Follow these steps if you need assistance

a. Remove the 4th argument that represented the number of elements

b. Remove the for loop while keeping the statement(s) inside it

c. At the beginning of the kernel, write the line of code that retrieves the current global index in the (0) dimension and assigned it to a variable i of type size_t

d. Use i to dereference the input and output arrays.

e. Save the file.

____ 6. If you closed the terminal

a. Open a terminal

b. cd to the **/home/student/fpga_trn/OpenCL/OCL_19_1/** folder

c. source opencl_init.sh

____ 7. If you didn't close the terminal, make sure you're in the following directory.

```
/home/student/fpga_trn/OpenCL/OCL_19_1/
```

____ 8. In the terminal, type the following command to compile the kernel and verify that it is error free.

```
aoc -march=emulator -fast-emulator -board=a10gx
SimpleKernel.cl
```

Step 2. Change the host code to launch a single work-item kernel

____ 1. Reopen **main.cpp** in Eclipse if it is not already open.

____ 2. Comment out the 4th kernel.setArg call that passes in the vectorSize

This is no longer needed

____ 3. Comment out the enqueueTask call

____ 4. Right below it, add the **enqueueNDRangeKernel** function call to launch the kernel in a multi-threaded fashion

Use the following steps if you need help

a. Global work offset should be cl::**NullRange**

b. Global work size is stored in the variable **vectorSize**

c. Work-item per workgroup is stored in the variable **workSize**

*With this command, you'll be launching the kernel **vectorSize** times. This is how data parallelism is achieved in OpenCL.*

____ 5. Save **main.cpp**

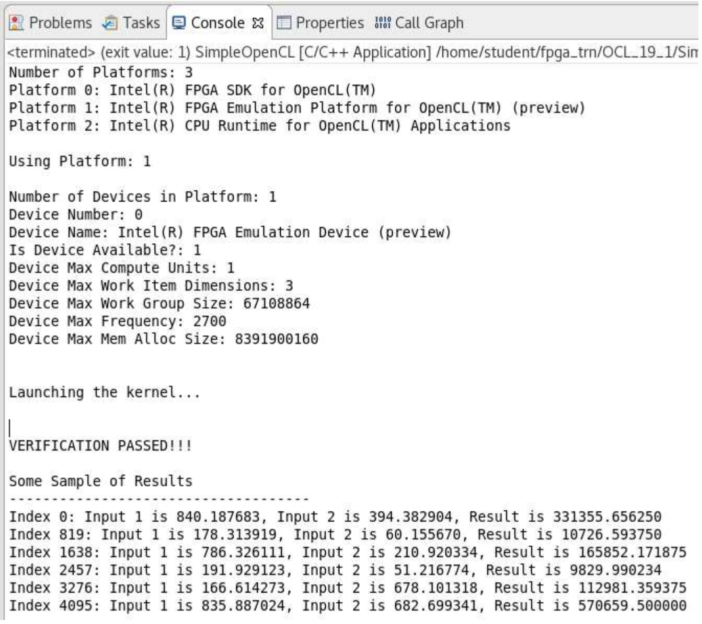
Step 3. Run and debug the kernel

____ 1. Compile the Project

____ 2. Run the program. Run or Debug the application.

____ 3. When the program runs to successful completion, you should see in the **Console** tab the “VERIFICATION PASSED!!!” message, along with some samples of results.

The Verification Passed message means the contents of Z and CalcZ are the same. You can also verify that the sample of results matches the math operation you’ve performed.



```
<terminated> (exit value: 1) SimpleOpenCL [C/C++ Application] /home/student/fpga_trn/OCL_19_1/Sin
Number of Platforms: 3
Platform 0: Intel(R) FPGA SDK for OpenCL(TM)
Platform 1: Intel(R) FPGA Emulation Platform for OpenCL(TM) (preview)
Platform 2: Intel(R) CPU Runtime for OpenCL(TM) Applications

Using Platform: 1

Number of Devices in Platform: 1
Device Number: 0
Device Name: Intel(R) FPGA Emulation Device (preview)
Is Device Available?: 1
Device Max Compute Units: 1
Device Max Work Item Dimensions: 3
Device Max Work Group Size: 67108864
Device Max Frequency: 2700
Device Max Mem Alloc Size: 8391900160


Launching the kernel...


VERIFICATION PASSED!!!

Some Sample of Results
------------------------------------
Index 0: Input 1 is 840.187683, Input 2 is 394.382904, Result is 331355.656250
Index 819: Input 1 is 178.313919, Input 2 is 60.155670, Result is 10726.593750
Index 1638: Input 1 is 786.326111, Input 2 is 210.920334, Result is 165852.171875
Index 2457: Input 1 is 191.929123, Input 2 is 51.216774, Result is 9829.990234
Index 3276: Input 1 is 166.614273, Input 2 is 678.101318, Result is 112981.359375
Index 4095: Input 1 is 835.887024, Input 2 is 682.699341, Result is 570659.500000
```

____ 4. Close Eclipse.

Exercise Summary

- Created an ND Range Kernel
- Launched the kernel using enqueueNDRangeKernel

Congratulations!

You have completed Lab 3

Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.