AMD **Accelerated** Parallel Processing TECHNOLOGY

SAMPLE

Image Binarization

1 Overview

1.1 Location

`$<APPSDKSamplesInstallPath>\samples\opencl\cl\`

1.2 How to Run

See the *Getting Started* guide for how to build samples. You first must compile the sample.

Use the command line to change to the directory where the executable is located. The pre-compiled sample executable is at `$<APPSDKSamplesInstallPath>\samples\opencl\bin\x86\` for 32-bit builds, and at `$<APPSDKSamplesInstallPath>\samples\opencl\bin\x86_64\` for 64-bit builds.

Ensure that the OpenCL 2.0 environment is installed.

Type the following command(s).

1. `ImageBinarization`
   This runs the program with the default options.
2. `ImageBinarization -h`
   This prints the help file.

1.3 Command Line Options

Table 1 lists, and briefly describes, the command line options.

Table 1 Command Line Options

Short Form	Long Form	Description
`-h`	`--help`	Shows all command options and their respective meanings.
	`--device`	Devices on which the program is to be run. Acceptable values are cpu or gpu.
`-q`	`--quiet`	Quiet mode. Suppresses most text output.
`-e`	`--verify`	Verify results against reference implementation.
`-t`	`--timing`	Print timing related statistics.
	`--dump`	Dump binary image for all devices.
	`--load`	Load binary image and execute on device.
	`--flags`	Specify compiler flags to build the kernel.
`-p`	`platformId`	Select platformId to be used (0 to N-1, where N is the number of available platforms).
`-d`	`deviceId`	Select deviceId to be used (0 to N-1, where N is the number of available devices).

Short Form	Long Form	Description
-v	version	AMD APP SDK version string.
-i	iterations	Number of iterations for kernel execution.
-x	globalThreadsHistogram	Global work-size for Histogram Kernel

2 Introduction

This sample demonstrates how to use images with read_write qualifier support, which is introduced in the OpenCL 2.0 OpenCL C programming language. This sample uses the `clCreateImage` API with the `CL_MEM_READ_WRITE` flag to create image objects, which are read and written by a kernel.

This sample must be run in the OpenCL 2.0 environment.

3 Implementation Details

The sample implements an Image Binarization algorithm, which converts a given gray-scale image into a binary image. The sample implements two kernels:

ImageHistogramKernel: This kernel calculates the histogram bins of a given gray-scale image. This histogram bins are then passed to the host. A threshold value is calculated on the host using Otsu's algorithm.

ImageBinarizationKernel: This kernel takes an image object and Otsu's threshold value as input. Each thread first reads a pixel and then changes the pixel value based on the threshold. Finally, the modified pixels are written back to the same image. (**Note:** This sample uses sampler read.).

4 References

1. The OpenCL Specification, version 2.0, rev 22 document.
2. The OpenCL C Programming Language (ver 2.0, rev 22) document.
3. http://en.wikipedia.org/wiki/Otsu's_method.
4. http://cis.k.hosei.ac.jp/~wakahara/otsu_th.c.

Contact

Advanced Micro Devices, Inc.
One AMD Place
P.O. Box 3453
Sunnyvale, CA, 94088-3453
Phone: +1.408.749.4000

For AMD Accelerated Parallel Processing:

URL: developer.amd.com/appsdk
Developing: developer.amd.com/
Forum: developer.amd.com/openclforum

The contents of this document are provided in connection with Advanced Micro Devices, Inc. (“AMD”) products. AMD makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication and reserves the right to make changes to specifications and product descriptions at any time without notice. The information contained herein may be of a preliminary or advance nature and is subject to change without notice. No license, whether express, implied, arising by estoppel or otherwise, to any intellectual property rights is granted by this publication. Except as set forth in AMD’s Standard Terms and Conditions of Sale, AMD assumes no liability whatsoever, and disclaims any express or implied warranty, relating to its products including, but not limited to, the implied warranty of merchantability, fitness for a particular purpose, or infringement of any intellectual property right.

AMD’s products are not designed, intended, authorized or warranted for use as components in systems intended for surgical implant into the body, or in other applications intended to support or sustain life, or in any other application in which the failure of AMD’s product could create a situation where personal injury, death, or severe property or environmental damage may occur. AMD reserves the right to discontinue or make changes to its products at any time without notice.

Copyright and Trademarks

© 2014 Advanced Micro Devices, Inc. All rights reserved. AMD, the AMD Arrow logo, ATI, the ATI logo, Radeon, FireStream, and combinations thereof are trademarks of Advanced Micro Devices, Inc. OpenCL and the OpenCL logo are trademarks of Apple Inc. used by permission by Khronos. Other names are for informational purposes only and may be trademarks of their respective owners.