

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 `$<APPSDKSamplesInstallPath>\samples\opencl\bin\x86\` for 32-bit builds, and `$<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. `FineGrainSVM`
This command runs the program with the default options.
2. `FineGrainSVM -h`
This command 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 [cpu gpu]	Devices on which the OpenCL kernel is to be run. Acceptable values are <code>cpu</code> or <code>gpu</code> .
-q	--quiet	Quiet mode. Suppresses all text output.
-e	--verify	Verify results against reference implementation.
-t	--timing	Print timing-related statistics.
-v	--version	AMD APP SDK version string.
	--dump [filename]	Dump the binary image for all devices.
	--load [filename]	Load the binary image and execute on the device.
	--flags [filename]	Specify the filename containing the compiler flags for building the kernel.
-i	--iterations	Number of iterations.
-p	--platformId	Select the platformId to be used[0 to N-1 where N is number platform s available].
-d	--deviceId	Select deviceId to be used[0 to N-1 where N is number devices available].
-x	--length	Length of the input array.

2 Introduction

This sample demonstrates the memory model of loads and stores in new C++11 standard, which is adopted by OpenCL 2.0. This sample creates two fine-grain buffers by the host. The kernel updates one of the buffers using atomics call (with relevant memory order and scope); the updates are available on the other fine grain SVM buffer too. This demonstrates an important concept of how updates are visible across devices in C11 memory model with atomics.

3 Implementation

This sample demonstrates the usage of 2.0 atomics and fine grain SVM. This sample creates two fine grain buffers on the host: one with atomics support, and the other with just SVM fine grain. The host updates the atomics buffer first and kernel waits for this update using atomics. In the second stage, the kernel updates the atomics buffer and host waits for this update and exits.

The sample workflow is as follows:

1. The host creates two fine grain buffers, `atomicsBuffer` and `buffer`. The `atomicsBuffer` is with atomics support.
2. The kernel waits for CPU to update the `atomicsBuffer` with 99.
3. The kernel updates the "buffer" with some value (+id) and then updates `atomicsBuffer` with (100+i) with release semantics.
4. The CPU waits for the updates and checks for the final values in the buffers.

This sample must be run in the OpenCL 2.0 environment.

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/
Support: developer.amd.com/appsdksupport
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.