AMD **Accelerated**
Parallel Processing
TECHNOLOGY

SAMPLE

OpenCL - DX9 Inter-operability

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 default executables are placed in $<APPSDKSamplesInstallPath>\samples\opencl\bin\x86 for 32-bit builds and $<APPSDKSamplesInstallPath>\samples\opencl\bin\x86_64\ for 64-bit builds.

Before building this sample:

- Ensure you have the DirectX SDK installed.

Type the following command(s).

1. SimpleDX9
   Produces the image that has the original image and the mirror image of the original image.

2. SimpleDX9 -h
   This command prints the help message.

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	Execute the OpenCL kernel on a device.
-q	--quiet	Quiet mode. Suppresses all text output.
-e	--verify	Verify results against reference implementation.
-t	-timing	Print timing.
	--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 the platform ID to be used (0 to N-1, where N is the number of available platforms).
-d	--deviceId	Select device ID to be used (0 to N-1, where N is the number of available devices).
-v	--version	AMD APP SDK version string.
-n	--numFrames	The number of frames.

2 Introduction

This sample demonstrates the usage of the the DX9 - OCL interop (`cl_khr_dx9_media_sharing`) extension.

This OpenCL extension allows applications to use media surfaces as OpenCL memory objects. It allows efficient sharing of data between OpenCL and selected adapter APIs. If this extension is supported, an OpenCL image object can be created from a media surface and the OpenCL API can be used to execute kernels that read and/or write memory objects that are media surfaces.

This sample helps to understand this API by giving an example of mirroring an image. The sample can be run on only Windows.

3 Implementation

The device must have support for the `cl_khr_dx9_media_sharing` extension.

The program reads an image from a file. It mirrors the image using OpenCL, then renders the processed image using DirectX 9.

You must have a graphics driver and an OpenCL runtime that support the Direct3D 9 media sharing extension. To verify that they are installed, run `clinfo` from the command line, and check if your device supports the `cl_dx9_media_sharing` extension. The interfaces for this extension will be provided in `cl_dx9_media_sharing.h`.

The following are the key functions of SimpleDX9:

1.   `clGetDeviceIDsFromDX9MediaAdapterKHR`: To query a media adapter for any associated OpenCL devices.

2. `clCreateFromDX9MediaSurfaceKHR`: Creates an OpenCL image object from a media surface.

3. `clEnqueueAcquireDX9MediaSurfacesKHR`: Acquire OpenCL memory objects that have been created from a media surface.

4. `clEnqueueReleaseDX9MediaSurfacesKHR`: Release OpenCL memory objects that have been created from media surfaces.

4 Algorithm

Input: `inputimage` - The input image given to the kernel. Size of `inputimage` is (w,h).

Output: `outputimage` - This image has both the input image and the mirrored image. Size of `outputimage` is (2w,h).

```
For each p(x,y) in  inputimag
 //where x varies from 1 to w and
 //y varies from 1 to h
     Write P (x,y) to  P1(x,y ) in outputimage
     Write P(x,y) to  P2(2w -x,y) in outputimage
```

5 References

1. http://www.khronos.org/registry/cl/specs/opencl-1.2-extensions.pdf

---

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**

AMD