SAMPLE

HeatPDE

1 Overview

1.1 Location

`$<AMDAPPSDKSamplesInstallPath>\samples\opencl\cl\2.0`

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 `$<AMDAPPSDKSamplesInstallPath>\samples\opencl\bin\x86_64\` for 64-bit builds. Ensure that the OpenCL 2.0 environment is installed.

Type the following command(s).

1. `HeatPDE`
   This command runs the program with the default options.
2. `HeatPDE -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 meaning.
	--device	Devices on which the program is to be run. Acceptable values are cpu or gpu.
-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 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).
-v	--version	AMD APP SDK version string.
-i	--iterations	Number of iterations for kernel execution.
-g	--gui	Display information on a GUI.

2 Introduction

This sample shows CPU and GPU co-ordination using SVM.

In this sample, a controlled heat field (governed by the heat equation) is simulated on a rectangular plate. It is assumed that the plate is heated by a grid of burners (placed at the spots marked by triangles in the graphical display). Heat sensors (placed at spots marked by rectangles in the graphical display) measure the heat at the spot they are fixed. Each heat sensor is given a minimum and a maximum temperature. A feedback loop monitors the temperatures at the heat sensors. If the temperature at a sensor goes below (above) its minimum (maximum) temperature, the feedback loop triggers its surrounding four burners on (off). The graphical display displays the heat field, sensor temperatures, and burner states (on/off).

This sample must be run in the OpenCL 2.0 environment.

3 Implementation Details

The heat field is governed by a parabolic partial differential equation called the heat equation. The partial differential equation (PDE) is as follows:

$$\nabla^2 \varphi = \frac{1}{a^2} \frac{\partial \varphi}{\partial t}$$

$\varphi(x_1 x_2 \dots x_{n\dots} t)$: Heat function/Temperature

$x_1 x_2 \dots x_{n\dots}$: Spatial co-ordinates

`t`: Time

`a`: Constant called the rate of diffusion based on the conductivity of the material

This heat field is kept in Fine Grain Shared Virtual Memory (SVM), and is accessible to both the CPU and the GPU. Based on initial conditions, boundary conditions and burner states, the GPU keeps computing and updating the heat field in the SVM. The CPU reads the heat field at sensor positions and runs the feedback loop to switch on (off) the burners.

4 References

1. Smith, Justin R., "The Design and Analysis of Parallel Algorithms", Oxford University Press, 1993.
2. The OpenCL Specification (ver 2.0, rev 22) document.
3. The OpenCL C Programming Language (ver 2.0, rev 22) document.

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