AMD **Accelerated** Parallel Processing TECHNOLOGY

SAMPLE

N-Body Simulation

1 Overview

1.1 Location

`$<AMDAPPSDKSamplesInstallPath>\samples\opencl\cl\1.x`

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 `$<AMDAPPSDKSamplesInstallPath>\samples\opencl\bin\x86\` for 32-bit builds, and `$<AMDAPPSDKSamplesInstallPath>\samples\opencl\bin\x86_64\` for 64-bit builds.

Type the following command(s).

1. `NBody`
   This runs with default options; p = 256.
2. `NBody -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 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.
`-x`	`--particles`	Number of particles.
`-i`	`--iterations`	Number of iterations for kernel execution.
`-g`	`--gui`	Enable graphical display

2 Introduction

An N-body simulation is a simulation of a large number of particles under the influence of physical forces such as gravity and coulomb forces. N-body simulations are used in various fields, including molecular dynamics, astrophysics, and Lagrangian mechanics.

The problem we seek to solve in this sample is a generalized problem of finding the collective motions of all the particles under the influence of forces between the bodies. This sample considers Newtonian forces between bodies and showcases the superior SIMD capabilities of AMD GPUs to solve N-body simulations in contrast to a CPU. A typical scenario is the motion of a large number of stars in a galaxy, starting from known initial conditions.

3 The N-Body Problem

This section provides a brief overview of the implemented N-body problem. The initial inputs to the problem are a set of n bodies, $b_1, b_2, \ldots b_n$, where each body b_i has a mass m_i, a velocity v_i, and position p_i. The distance between any two bodies, b_i and b_j, is written d_{ij}, we want to compute the new positions and velocity of each body after dT time elapsed and with softening factor f_c.

To calculate the new velocity and position

1. Calculate acceleration, a_i, effect on b_i due to each body, b_j:

$$a_i = \sum_{j=0}^{j=n} s * d_{ij}$$

where d_{ij} is distance between b_j and b_i.

The s in the previous equation is calculated as follows:

$$s = m_i * \left(1 \Big/ \sqrt{f_c * \overline{d_{ij}}} \right)$$

Here: m_i is mass of b_i
f_c is softening factor
$\overline{d_{ij}}$ is the magnitude of distance vector

2. Calculate the new position and velocity from initial position (p_i), velocity (v_i) and calculated acceleration (a_i):

$$p_i = p_i + v_i * dT + a_i * 0.5 * (dT)^2$$
$$v_i = v_i + a_i * dT$$

4 Implementation Details

Each work item calculates the position and velocity from the equation in 1, above. Shared memory is used to reduce memory bandwidth and to reuse data.

5 Recommended Input Option Settings

For best performance, enter the following on the command line: `-x 32768 -i 10 -q -t`

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