SAMPLE

PrefixSum

1 Overview

1.1 Location

`$(AMDAPPSDKSAMPLESROOT)\samples\opencl\cl\app`

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 precompiled sample executable is at `$(AMDAPPSDKSAMPLESROOT)\samples\opencl\bin\x86\` for 32-bit builds, and `$(AMDAPPSDKSAMPLESROOT)\samples\opencl\bin\x86_64\` for 64-bit builds.

Type the following command(s).

1. `PrefixSum`
   Runs with default options; x = 128.
2. `PrefixSum -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	--length	Length of the input array.
-i	--iterations	Number of iterations for kernel execution.

2 Introduction

In Computer Science, the prefix sum, scan, or cumulative sum of a sequence of numbers x0, x1, x2, ... is a second sequence of numbers y0, y1, y2, ..., the sums of prefixes of the input sequence:

```
y0 = x0
y1 = x0 + x1
y2 = x0 + x1+ x2
...
```

Prefix sums are trivial to compute in sequential models of computation, by using the formula `yi = yi - 1 + xi` to compute each output value in sequence order.

2.1 Parallel implementation

A prefix sum can be calculated in parallel by performing the following steps.

1. Compute the sums of consecutive pairs of items in which the first item of the pair has an even index: z0 = x0 + x1, z1 = x2 + x3, etc.
2. Recursively compute the prefix sum w0, w1, w2, ... of the sequence z0, z1, z2, ...
3. Expand each term of the sequence w0, w1, w2, ... into two terms of the overall prefix sum: y0 = x0, y1 = w0, y2 = w0 + x2, y3 = w1, etc. After the first value, each successive number yi is either copied from a position half as far through the w sequence, or is the previous value added to one value in the x sequence.

If the input sequence has n steps, then the recursion continues to a depth of O(log n), which is also the bound on the parallel running time of this algorithm. The number of steps of the algorithm is O(n), and it can be implemented on a parallel random access machine with O(n/log n) processors without any asymptotic slowdown by assigning multiple indices to each processor in rounds of the algorithm for which there are more elements than processors.

Parallel algorithms for prefix sums can often be generalized to other scan operations on associative binary operations, and they can also be computed efficiently on modern parallel hardware such as a GPU. Many parallel implementations follow a two pass procedure where partial prefix sums are calculated in the first pass on each processing unit; the prefix sum of these partial sums is then calculated and broadcast back to the processing units for a second pass using the now known prefix as the initial value. Asymptotically this method takes approximately two read operations and one write operation per item.

3 References

1. http://en.wikipedia.org/wiki/Prefix_sum

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