

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.

Type the following command(s).

1. `MatrixTranspose`
Transforms an arbitrary array of 64.
2. `MatrixTranspose -h`
This 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 meaning.
	--device	Devices on which the program 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.
	--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	--width	Width/height dimension of the square matrix.
-b	--blockSize	Use local memory of dimensions blockSize x blockSize.
-i	--iterations	Number of iterations for kernel execution.

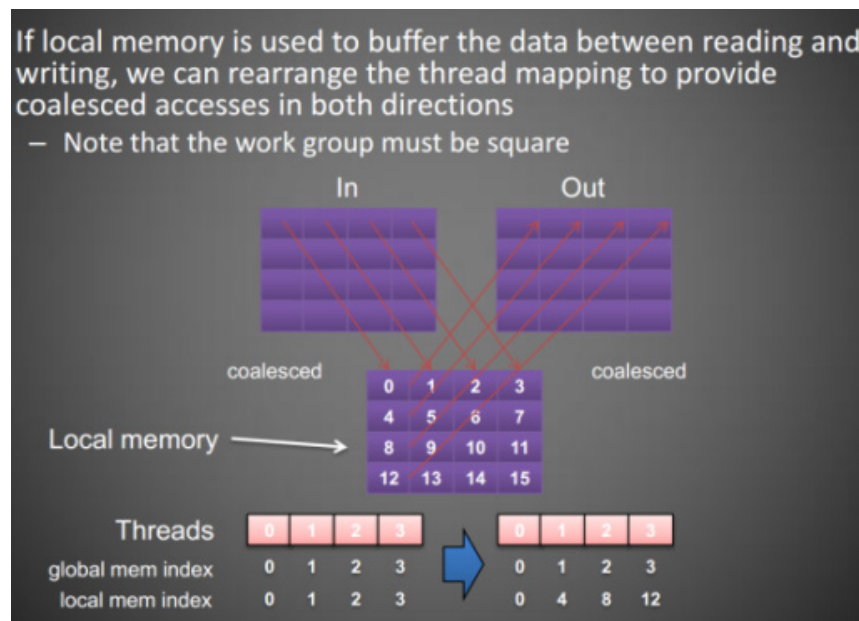
2 Introduction

In linear algebra, the transpose of a matrix A is another matrix A^T (also written A' , A^{tr} or A^{t}). Formally, the i^{th} row, j^{th} column element of A^T is the j^{th} row, i^{th} column element of A .

If A is an $m \times n$ matrix then A^T is an $n \times m$ matrix.

3 Implementation

The implementation takes by default the workgroup of size 256 in 16×16 format. If the first row of work-items reads the input matrix in a coalesced manner, the elements read must be written into different rows but into the same column. Such an implementation will cause global memory channel conflicts and will be very slow.



To handle this situation, Local Data Share (LDS) is used as an intermediary buffer. A workgroup puts a block of memory from the input matrix into this LDS block. This write can occur in a coalesced manner. The elements of the block are transposed inside the LDS. Even this approach will cause LDS bank conflicts, but the conflicts are far less severe. The LDS block can now be written back to the global memory in a coalesced manner, as the matrix is already transposed.

4 References

1. <http://en.wikipedia.org/wiki/Transpose>
2. <https://www.cvg.ethz.ch/teaching/2011spring/gpgpu/GPU-Optimization.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



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

© 2013 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.