CUDA DEMO SUITE

TRM-06704-001_v11.0 | August 2020

Reference Manual

TABLE OF CONTENTS

Chapter 1. INTRODUCTION

The CUDA Demo Suite contains pre-built applications which use CUDA. These applications demonstrate the capabilities and details of NVIDIA GPUs.

Chapter 2. DEMOS

Below are the demos within the demo suite.

2.1. deviceQuery

This application enumerates the properties of the CUDA devices present in the system and displays them in a human readable format.

2.2. vectorAdd

This application is a very basic demo that implements element by element vector addition.

2.3. bandwidthTest

This application provides the memcopy bandwidth of the GPU and memcpy bandwidth across PCI-e. This application is capable of measuring device to device copy bandwidth, host to device copy bandwidth for pageable and page-locked memory, and device to host copy bandwidth for pageable and page-locked memory.

Arguments:

```
Usage:  bandwidthTest [OPTION]...
Test the bandwidth for device to host, host to device, and device to device
 transfers

Example:  measure the bandwidth of device to host pinned memory copies in the
 range 1024 Bytes to 102400 Bytes in 1024 Byte increments
./bandwidthTest --memory=pinned --mode=range --start=1024 --end=102400 --
increment=1024 --dtoh
```

Options	Explanation
--help	Display this help menu

Options	Explanation
--csv	Print results as a CSV
--device=[deviceno] all 0,1,2,...,n	Specify the device device to be used compute cumulative bandwidth on all the devices Specify any particular device to be used
--memory=[MEMMODE] pageable pinned	Specify which memory mode to use pageable memory non-pageable system memory
--mode=[MODE] quick range shmoo	Specify the mode to use performs a quick measurement measures a user-specified range of values performs an intense shmoo of a large range of values
--htod	Measure host to device transfers
--dtoh	Measure device to host transfers
--dtod	Measure device to device transfers
--wc	Allocate pinned memory as write-combined
--cputiming	Force CPU-based timing always
Range Mode options --start=[SIZE] --end=[SIZE] --increment=[SIZE]]	 Starting transfer size in bytes Ending transfer size in bytes Increment size in bytes

2.4. busGrind

Provides detailed statistics about peer-to-peer memory bandwidth amongst GPUs present in the system as well as pinned, unpinned memory bandwidth.

Arguments:

Options	Explanation
-h	print usage
-p [0,1]	enable or disable pinned memory tests (default on)
-u [0,1]	enable or disable unpinned memory tests (default off)
-e [0,1]	enable or disable unpinned memory tests (default off)
-u [0,1]	enable or disable p2p enabled memory tests (default on)
-d [0,1]	enable or disable p2p disabled memory tests (default off)

Options	Explanation
-a	enable all tests
-n	disable all tests

```
Order of parameters maters.
Examples:
    ./BusGrind -n -p 1 -e 1   Run all pinned and P2P tests
    ./BusGrind -n -u 1        Runs only unpinned tests
    ./BusGrind -a             Runs all tests (pinned, unpinned, p2p enabled, p2p
 disabled)
```

2.5. nbody

This demo does an efficient all-pairs simulation of a gravitational n-body simulation in CUDA. It scales the n-body simulation across multiple GPUs in a single PC if available. Adding "-numbodies=num_of_bodies" to the command line will allow users to set # of bodies for simulation. Adding "-numdevices=N" to the command line option will cause the sample to use N devices (if available) for simulation. In this mode, the position and velocity data for all bodies are read from system memory using "zero copy" rather than from device memory. For a small number of devices (4 or fewer) and a large enough number of bodies, bandwidth is not a bottleneck so we can achieve strong scaling across these devices.

Arguments:

Options	Explanation
-fullscreen	run n-body simulation in fullscreen mode
-fp64	use double precision floating point values for simulation
-hostmem	stores simulation data in host memory
-benchmark	run benchmark to measure performance
-numbodies=N	number of bodies (>= 1) to run in simulation
-device=d	where d=0,1,2.... for the CUDA device to use
-numdevices=i	where i=(number of CUDA devices > 0) to use for simulation
-compare	compares simulation results running once on the default GPU and once on the CPU
-cpu	run n-body simulation on the CPU
-tipsy=file.bin	load a tipsy model file for simulation

2.6. oceanFFT

This is a graphical demo which simulates an ocean height field using the CUFFT library, and renders the result using OpenGL.

The following keys can be used to control the output:

Keys	Function
w	Toggle wireframe

2.7. randomFog

This is a graphical demo which does pseudo- and quasi- random numbers visualization produced by CURAND. On creation, randomFog generates 200,000 random coordinates in spherical coordinate space (radius, angle rho, angle theta) with curand's XORWOW algorithm. The coordinates are normalized for a uniform distribution through the sphere. The X axis is drawn with blue in the negative direction and yellow positive. The Y axis is drawn with green in the negative direction and magenta positive. The Z axis is drawn with red in the negative direction and cyan positive.

The following keys can be used to control the output:

Keys	Function
s	Generate new set of random nos and display as spherical coordinates (Sphere)
e	Generate new set of random nos and display on a spherical surface (shEll)
b	Generate new set of random nos and display as cartesian coordinates (cuBe/ Box)
p	Generate new set of random nos and display on a cartesian plane (Plane)
i, l, j	Rotate the negative Z-axis up, right, down and left respectively
a	Toggle auto-rotation
t	Toggle 10x zoom
z	Toggle axes display
x	Select XORWOW generator (default)
c	Select Sobol' generator
v	Select scrambled Sobol' generator
r	Reset XORWOW (i.e. reset to initial seed) and regenerate
]	Increment the number of Sobol' dimensions and regenerate
[	Reset the number of Sobol' dimensions to 1 and regenerate

Keys	Function
+	Increment the number of displayed points by 8,000 (max. 200,000)
-	Decrement the number of displayed points by 8,000 (down to min. 8000)
q/[ESC]	Quit the application.

Notice

Trademarks

Copyright