Fountainhead

CUDA Memory Model

~ Global, Constant, Shared, Texture ~

Andrew Sheppard

Baruch College MFE “Big Data in Finance” Course

30th January 2014

Fountainhead

Objectives

In this talk I will cover:

1. Differences between CPU and GPU memory models.
2. CUDA global memory.
3. CUDA constant memory.
4. CUDA registers and shared memory.
5. CUDA texture memory.
6. CUDA advanced memory topics.

Fountainhead

Terminology

New terminology introduced in this talk:

Memory	Location	Cached	Access	Who
Local	Off-chip	No	Read/Write	One Tthread
Shared	On-chip	N/A	Read/write	All threads in a block
Global	Off-chip	No	Read/write	All threads + CPU
Constant	Off-chip	Yes	Read	All threads + CPU
Texture	Off-chip	Yes	Read	All threads + CPU

Fountainhead

~ 1. CPU versus GPU ~

CPU (host) memory model:

- Latency driven.
- Lots of cache (256KB L2, 8MB L3 - typical).
- Data bus width 256-bits.
- 3.2 GHz clock, 25 GB/s data transfer rate.

Fountainhead

CPU versus GPU (cont.)

GPU (device) memory model:

- Throughput driven.
- Some cache (256KB L2, 8MB L3 - typical).
- Data bus width 512-bits. 16 consecutive 32-bit words.
- 1.15 GHz clock, 144 GB/s data transfer rate to global memory, 1 TB/s data transfer rate from shared memory to cores.

Fountainhead

~ 2. Global Memory ~

Global memory:

- Large (up to 6GB on current GPUs).
- Global memory fetch from a thread may take 100's of clock cycles.

Fountainhead

~ 3. Constant Memory ~

All ideas have a beginning and here's how this idea came About …

Fountainhead

~ 4. Registers & Shared Memory ~

Shared memory:

- Used for inter-thread shared data and communication.
- Small and low latency. “Register” speed.
- High bandwidth.
- Limited size (Fermi 64KB, configurable as 16KB L1 cache/48KB shared memory, or vice versa).

Fountainhead

~ 5. Texture Memory ~

All ideas have a beginning and here's how this idea came About …

Fo

~ ~

All ideas h idea came
About …