CUDA Introduction

Piotr Luszczek
Per-Core Performance

Individual core speed no longer increases
GPU vs. CPU Performance over Years

- **GPU FP32 GeForce GTX**
- **GPU FP64 Tesla**
- **CPU FP32**
- **CPU FP64**

**Graphics-oriented GPUs**

**Compute-oriented GPUs**

- Core speed
- Core count

Year

- 2002
- 2003
- 2004
- 2005
- 2006
- 2007
- 2008
- 2009
- 2010
- 2011
- 2012

Peak (Gflops/s)
GPU and GPGPU: Origin Story

- Programmable graphics pipeline
  - GLSL
- Interpolation vs. dynamic range
  - Colors in graphics look better in floating-point
- Early attempts at programming
  - Cg, Brook, …
- Modern standards or de facto standards
  - CUDA (currently 8)
    - Compute Unified Device Architecture
  - OpenCL (currently 2)
- High-level languages
  - OpenMP 4
  - OpenACC
Hardware: CPU vs. GPU

Main Memory RAM: DDR3 or DDR4
Size: ~100 GiB
Speed: ~50 GB/s

PCIexpress

GPU Memory RAM: GDDR5
Size: ~10 GiB
Speed: ~200 GB/s

Long latency
Software: CPU + GPU
__global__ void sum(double x, double y, double *z) {
    *z = x + y;
}

int main(void) {
    double *device_z;

cudaMalloc( &device_z, sizeof(double) );

sum<<<1,1>>>(2, 3, device_z);

cudaMemcpy( &host_z, device_z, sizeof(double), cudaMemcpyDeviceToHost);

printf("%g\n", host_z);

cudaFree(device_z);

return 0;
}

$ nvcc sum.cu -o sum
$ ./sum
5