Index

1. GPU & CUDA
   - What is GPU?
   - What is SIMD?
   - What is CUDA?
   - Simple example

2. About assignment...
GPGPU & CUDA
What is GPU?

- **GPU**
  - **Graphic Processing Unit**
    - A specialized circuit designed to rapidly manipulate data to accelerate the building of images in a frame buffer
    - Their highly *parallel* structure makes them more effective than general-purpose CPUs for algorithms where processing of large blocks of data is done *in parallel*

- **GPGPU**
  - **General-purpose computing on GPU**
    - The means of using a GPU, which typically handles computation only for computer graphics, to perform computation *in applications traditionally handled by the CPU*
What is GPU?

- Comparison: CPU vs. GPU
What is GPU?

- **Comparison: CPU vs. GPU**

- **CPU**
  - CPU 0
  - CPU 1
  - CPU 2
  - CPU 3
  - L2 Cache

- **GPU**
  - SIMD
  - SIMD
  - SIMD
  - SIMD
  - SIMD
  - SIMD
  - SIMD
  - SIMD
  - SIMD
  - SIMD
  - SIMD
  - SIMD
  - SIMD
  - SIMD
  - SIMD
  - SIMD
  - SIMD
  - L2 Cache

- **Single Instruction Multiple Data (SIMD)**
Parallel execution in CPU

Fragment 1

Fragment 2

Source code (instructions)
Parallel execution in GPGPU

Sixteen cores: 16 simultaneous instruction streams

Be able to share an instruction stream
SIMD processing

- **SIMD**
  - Single Instruction Multiple Data
    - A class of parallel computers
    - Multiple processing elements that *perform the same operation on multiple data simultaneously*
  - Modern **GPUs** are often wide **SIMD** implementation
SIMD example

- Matrix multiplication

Same operation

Different data
What is CUDA?

- **CUDA**
  - **Compute Unified Device Architecture**
    - A parallel computing platform and programming model created by NVIDIA
    - It is accessible to software developers through CUDA-accelerated libraries, and extensions to C, C++ and Fortran.
    - C/C++ programmers use 'CUDA C/C++' (C/C++ with CUDA extensions to express parallelism, data locality, and thread cooperation, as well as some restrictions), compiled with "nvcc", to code algorithms for execution on the GPU.
What is CUDA?

- CUDA
  - Compute Unified Device Architecture

Processing flow on CUDA
Simple Example (C to CUDA conversion)

```c
__global__ void ForceCalcKernel(int nbodies, struct Body *body, ..) { ... }
__global__ void AdvancingKernel(int nbodies, struct Body *body, ...){ ... }

int main(...) {
    Body *body, *body1;
    ...
    cudaMemcpy((void**)&body1, sizeof(Body)*nbodies, cuda_HostToDevice);
    cudaMemcpy(body1, body, sizeof(Body)*nbodies, cudaMemcpyHostToDevice);
    for(timestep = ...) {
        ForceCalcKernel<<<1, 1>>>(nbodies, body1, ...);
        AdvancingKernel<<<1, 1>>>(nbodies, body1, ...);
    }
    cudaMemcpy(body, body1, sizeof(Body)*nbodies, cudaMemcpyDeviceToHost);
    cudaFree(body1);
    ...
}
```

Indicates GPU kernel that CPU can call
Allocate memory on GPU
Copy CPU data to GPU
Call GPU kernel with 1 block and 1 thread per block
Copy GPU data back to CPU
Advanced programming properties

- Advanced!!
  - Thread block architecture
    - Thread model
    - Grid block model
    - Thread scheduling
      - warp
  - Memory architecture
    - Memory architecture(layer)
    - Bank conflict
    - Memory coalescing
  - ...
  - ...

More Performance improvement
GPGPU & CUDA References

- http://ca.olin.edu/2008/gpgpu/index.htm
- http://vsts2010.net/205
Assignment...
CUDA assignment

- **DES (Data Encryption Standard) algorithm**
  - Make the C program which implements DES
    - Sequential execution
  - Make the CUDA program which implements DES
    - Parallel execution
  - Compare the result (execution time)
    - You should show that the parallel execution is faster!!

- Due date: the end of this semester (mid of December)
- Detailed specification will be on the class board
CUDA assignment

- DES (Data Encryption Standard) algorithm
  - An algorithm for the encryption of electronic data
  - Block cipher
    - It takes a fixed-length string of plaintext bits and transforms it through a series of complicated operations into another ciphertext bitstring of the same length
    - In the case of DES, the block size is 64 bits
  - DES also uses a key to customize the transformation
    - only be performed by those who know the particular key used to encrypt.
    - The key ostensibly consists of 64 bits; however, only 56 of these are actually used by the algorithm.
    - Eight bits are used solely for checking parity, and are thereafter discarded.
  - Two modes: ECB & CBC
    - The ECB is the way in which each block is encrypted independently
CUDA assignment

- DES (Data Encryption Standard) algorithm
DES References

- http://marcof.tistory.com/58#
QnA?

mysung@dcslab.snu.ac.kr

bye bye~ 😊
CUDA assignment

- DES (Data Encryption Standard) algorithm
GF100 architecture