Processamento Paralelo Matricial

CUDA

Fernando Leichtweis

Introdução

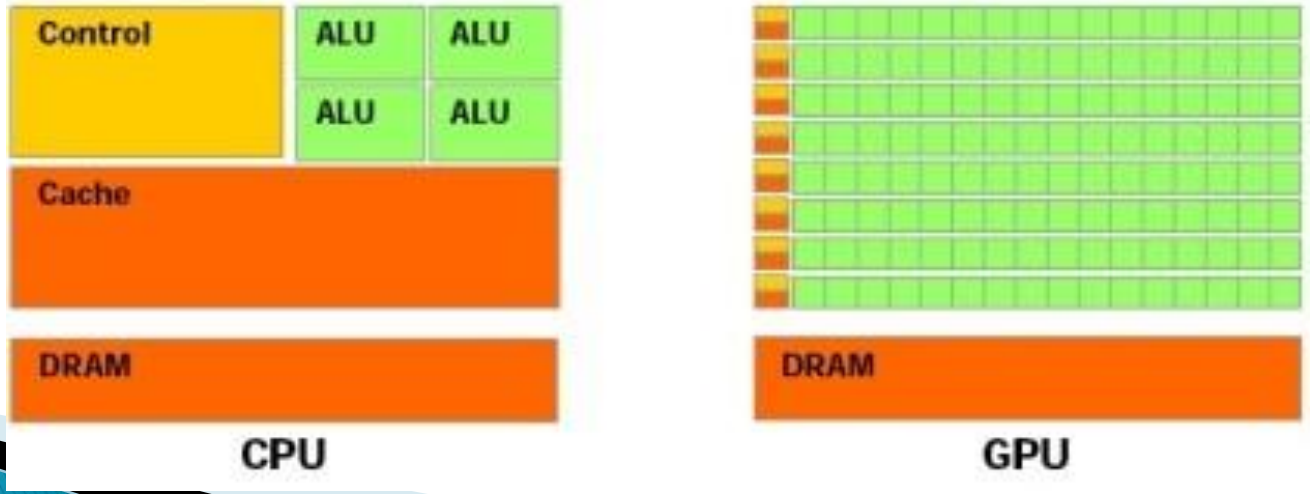
- CUDA™ é uma plataforma de computação paralela e um modelo de programação inventados pela NVIDIA. Ela permite aumentos significativos de performance computacional ao aproveitar a potência da unidade de processamento gráfico (GPU).

Introdução

- O modelo de programação CUDA foi projetado para expor todo o potencial de processamento paralelo das GPUs, possibilitando o desenvolvimento de aplicações que podem explorar ao máximo o paralelismo de dados.

Introdução

- Em uma dada aplicação, um mesmo trecho de código é executado em paralelo para pequenos blocos de dados, com a existência de várias pequenas caches e níveis de hierarquia de memória que escondem a latência de acesso a estes blocos.



Introdução

- CUDA é composta de três abstrações principais: uma hierarquia de grupo de threads, memórias compartilhadas e sincronização via barreiras.

- O CUDA possibilitou o acesso a memória de maneira a suportar operações antes somente suportadas por CPU's, como as operações de *gather* e *scatter* com a memória DRAM da GPU.

Aplicações

- Com milhões de GPUs habilitadas para CUDA já vendidas até hoje, os desenvolvedores de software, cientistas e pesquisadores estão descobrindo usos amplamente variados para a computação com GPU CUDA.
- Alguns exemplos de aplicações:

Aplicações

- Identificação de placas ocultas em artérias

  Ataques cardíacos causam um grande número de mortes no mundo todo. A Harvard Engineering, a Harvard Medical School e o Brigham & Women's Hospital se reuniram para usar GPUs com o objetivo de simular o fluxo sanguíneo e identificar placas arteriais ocultas, sem fazer uso de técnicas de imagem invasivas ou cirurgias exploratórias.

Aplicações

- Análise do fluxo de tráfego aéreo

  O National Airspace System (Sistema de Espaço Aéreo Nacional) gerência a coordenação do fluxo de tráfego aéreo em âmbito nacional. Modelos computacionais ajudam a identificar novas maneiras de aliviar congestionamentos e manter o tráfego de aeronaves fluindo de forma eficiente. Utilizando o poder computacional das GPUs, uma equipe da NASA obteve grande ganho de performance, reduzindo o tempo de análise de dez minutos para três segundos.

Aplicações

- POPULAR GPU-ACCELERATED APPLICATIONS CATALOG

http://www.nvidia.com/docs/IO/123576/nv-applications-catalog-lowres.pdf

Linguagens Suportadas

- Para escrever código utilizando CUDA, o programador pode utilizar C, C++ e Fortran.
- Para tornar possível a expressão do modelo de programação, o CUDA adiciona diversos marcadores especiais a estas linguagens. Estes marcadores, junto com o conjunto de funções fornecidas pela API, completam o que é necessário para desenvolver aplicativos utilizando CUDA.

CUDA C

Standard C Code

```
void saxpy_serial(int n,
                  float a,
                  float *x,
                  float *y)
{

  for (int i = 0; i < n; ++i)
    y[i] = a*x[i] + y[i];
}

// Perform SAXPY on 1M elements
saxpy_serial(4096*256, 2.0, x, y);
```

Parallel C Code

```
__global__
void saxpy_parallel(int n,
                    float a,
                    float *x,
                    float *y)
{
  int i = blockIdx.x*blockDim.x +
          threadIdx.x;
  if (i < n) y[i] = a*x[i] + y[i];
}

// Perform SAXPY on 1M elements
saxpy_parallel<<<4096,256>>>(n,2.0,x,y);
```

http://developer.nvidia.com/cuda-toolkit

Referências

- CUDA EM AMBIENTES DE *CLUSTER* E *CLOUD*, *Prof. Dr. Luciano Silva.* ERAD-SP, Campinas (SP), Julho de 2012.
- https://developer.nvidia.com/category/zone/cuda-zone, acessado em agosto de 2013.
- Lopes, B. C.; Azevedo, R. J. – Computação de alto desempenho utilizando CUDA. Instituto de Computação – Universidade Estadual de Campinas (Unicamp). Campinas/São Paulo.
- http://johntortugo.wordpress.com/2008/12/30/cuda-modelo-de-programacao-paralela/, acessado em agosto de 2013.

Processamento Paralelo Matricial

CUDA

Fernando Leichtweis