

Cours GPU master

Mercredi 6 Décembre



Équipes Traitement
de l'Information
et Systèmes



Partie 1

Introduction to GPU
Technology

Overview of GPUs

GPU vs CPU

Evolution of GPUs
to GPGPU

Applications

Partie 2

Architecture of GPU and GPU
Coding Paradigm

GPU Architecture

GPGPU & CUDA

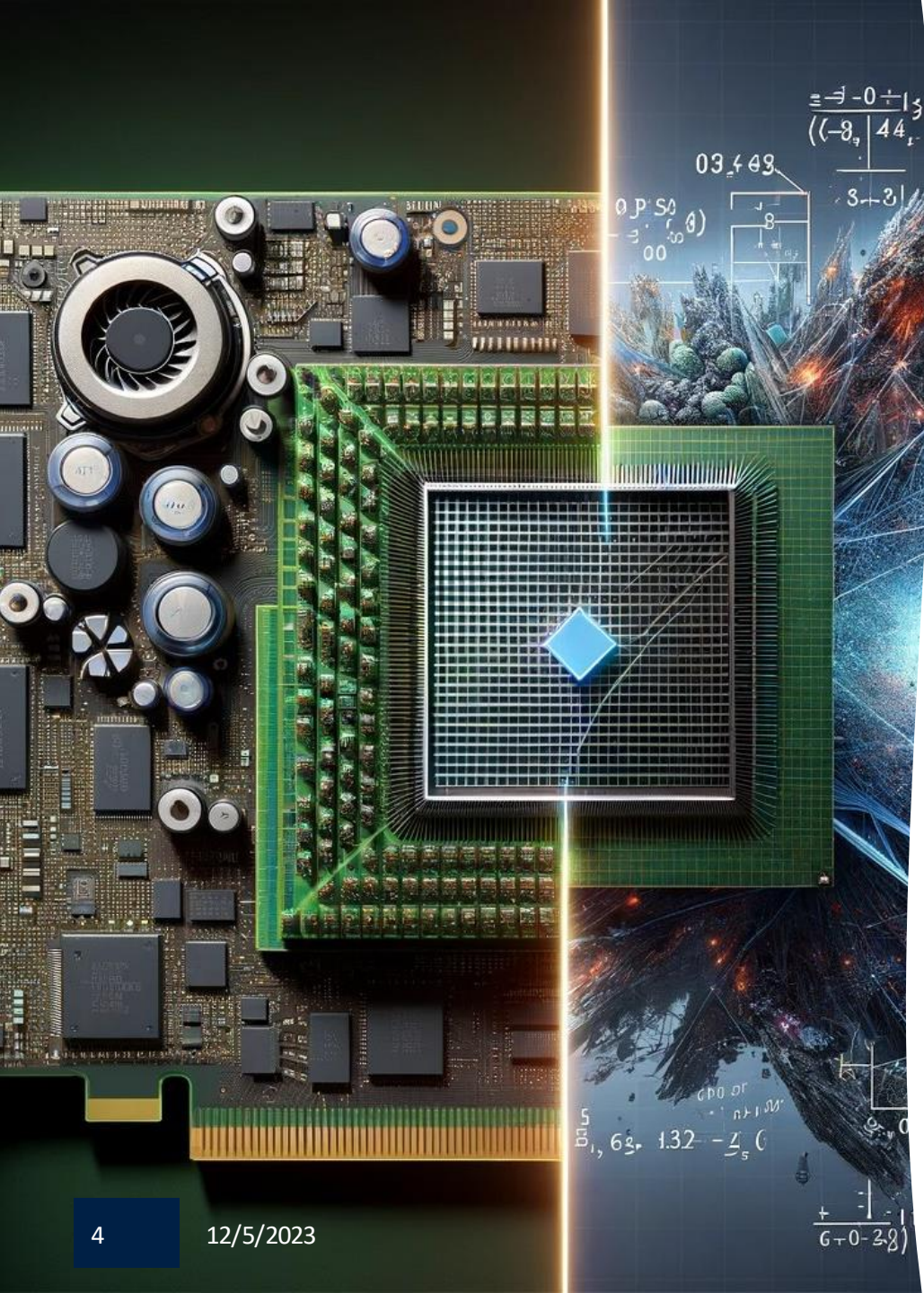
SIMT & Basic GPU
Programming



Infos et points à
évoquer

Partie 1.

Introduction to GPU Technology



• Overview of GPUs

- GPUs are specialized electronic circuits designed to rapidly manipulate and alter memory to accelerate the creation of images in a frame buffer intended for output to a display device.
- Originally designed for rendering graphics and images, GPUs have evolved to become highly efficient at mathematical calculations, especially those that can be performed in parallel.

Historical Development of GPUs & Role of GPUs in Modern Computing Systems

- The evolution of GPUs can be traced back to the 1970s and 1980s with the advent of arcade machines and home consoles.
- In the 1990s, GPUs became common in personal computers, marking the beginning of GPUs as separate processing units.
- The early 2000s saw the introduction of programmable shaders, allowing for more flexibility in graphics rendering.
- More recently, GPUs have been recognized for their ability to handle parallel processing tasks in non-graphics applications, giving rise to GPGPU (General-Purpose computing on Graphics Processing Units).
- GPUs play a critical role in modern computing, handling complex tasks like 3D rendering, video processing, and gaming.
- They have become essential in scientific research, cryptocurrency mining, deep learning, and artificial intelligence due to their efficient parallel processing capabilities.

GPU vs CPU

- CPUs are designed to handle a wide range of computing tasks and are optimized for sequential processing with a few cores capable of executing complex instructions.
- GPUs, in contrast, have hundreds or thousands of smaller cores designed for parallel processing, making them efficient at handling multiple similar operations concurrently.
- GPUs excel in applications where the same operation is performed on many data elements simultaneously (such as matrix operations or pixel rendering).
- CPUs are better suited for tasks that require complex decision-making and a variety of instructions, like running an operating system or executing diverse application code.

Examples of Tasks Better Suited for GPUs and Those Better for CPUs

- GPUs are ideal for graphics rendering, video editing, deep learning model training, and large-scale scientific computations.
- CPUs are better for general-purpose computing tasks, including running most software applications, managing system operations, and performing tasks that require diverse instructions and low latency.

Evolution of GPUs to GPGPU

- The shift towards GPGPU began as researchers and programmers started utilizing the parallel processing power of GPUs for non-graphical calculations.
- This was facilitated by the development of new programming models and languages (like CUDA and OpenCL) that allowed direct access to the GPU's instruction set and memory.
- Parallel processing involves dividing a large problem into smaller ones that can be solved concurrently.
- GPUs, with their large number of cores, are inherently designed for parallel execution, making them suitable for tasks that can be divided into multiple parallel operations.

Overview of GPGPU and Its Significance in Current Technology Trends

- GPGPU stands for General-Purpose Computing on Graphics Processing Units, a technique that utilizes GPUs for computation other than graphics.
- GPGPU has become increasingly important in fields requiring massive parallelism, such as machine learning, scientific simulation, and real-time data analysis.

Applications

- GPU:
 - 1. **Gaming**: Rendering complex 3D graphics in real time.
 - 2. **Graphic Design**: Powering software for image processing, video editing, and digital art creation.
 - 3. **Virtual Reality (VR)**: Providing the necessary computational power to create immersive VR environments.
 - 4. **Professional Visualization**: Used in CAD (Computer-Aided Design), CGI (Computer-Generated Imagery), and other visualization tasks where high-resolution, detailed images are crucial.
- GPGPU
 - Applications 1. **Scientific Computing**: Accelerating simulations and computations in fields like physics, chemistry, and biology.
 - 2. **Machine Learning and AI**: Training and deploying neural networks and other machine learning models.
 - 3. **Data Analysis**: Speeding up big data processing tasks and complex calculations in data science.
 - 4. **Cryptography**: Used in cryptocurrency mining and encryption/decryption processes.
 - 5. **Medical Imaging**: Assisting in processing and analyzing medical images such as MRI and CT scans.
 - 6. **Weather Forecasting**: Simulating climate and weather patterns for forecasting.
 - 7. **Oil and Gas Exploration**: Processing seismic data to locate oil and gas reserves. The versatility of GPUs and GPGPUs is primarily due to their ability to handle parallel processing efficiently, making them ideal for tasks that involve handling large amounts of data or performing complex computations simultaneously.

Partie 2.

Architecture of GPU and GPU Coding Paradigm

GPU Architecture

Slide INRIA

GPGPU & CUDA

Slide Nvidia

CUDA Paradigm

Rappel :

- Kernel : GPU function
- Threads : Execute the Kernel
- Warp – Group of 32 Threads -> execute 1 warp instruction
- Thread Block – Programmer deffined group of thread
- Grid – Group of Block



CUDA Paradigm

Rappel :



$$\text{Global TID} = \text{Blockidx.x} * \text{Blockdim.x} + \text{Threadidx.x}$$



Équipes Traitement
de l'Information
et Systèmes

Laboratoire Etis
6 Rue du Ponceau
95000 Cergy

etis-lab.fr

T.07 61 76 91 47

