Universidade de Brasília

Faculdade do Gama - FGA

Programação para Sistemas Paralelos e Distribuídos

Turma A 2023/1

Estudo extraclasse

Arquitetura de CPUs e GPUs

Autores:

Gabriela da Gama Pivetta - 180052845

Murilo Gomes de Souza - 180025601

Brasília

12 de junho de 2023

Sumário

1 Introdução

A evolução da tecnologia de processamento tem sido um elemento fundamental para impulsionar o avanço e a inovação em diversas áreas. No campo da computação, as arquiteturas de CPUs (Unidades de Processamento Central) e GPUs (Unidades de Processamento Gráfico) desempenham um papel crucial na execução de tarefas complexas. Este documento tem como objetivo fornecer uma visão geral sobre essas arquiteturas, explorando seu histórico, evoluções, aplicações e outras características em nível mais geral.

Ao longo das últimas décadas, a arquitetura de CPUs tem passado por um constante aprimoramento, impulsionado pelo rápido avanço da tecnologia de semicondutores e pela demanda crescente por maior desempenho computacional. As CPUs são responsáveis por executar as instruções de um programa, processando e manipulando os dados necessários para a realização de uma ampla variedade de tarefas. Desde os primeiros computadores, as CPUs têm evoluído em termos de velocidade de clock, capacidade de processamento paralelo, tamanho físico e eficiência energética.

Por outro lado, as GPUs surgiram inicialmente como componentes especializados em processamento gráfico, destinados a acelerar a renderização de imagens e a execução de tarefas relacionadas à computação gráfica. No entanto, com o tempo, as GPUs também têm sido utilizadas em aplicações gerais que exigem grande poder de processamento paralelo, como aprendizado de máquina, simulações científicas e mineração de criptomoedas. Sua arquitetura altamente paralela e capacidade de processamento massivamente distribuído tornaram as GPUs uma ferramenta valiosa para diversas áreas.

Ao abordar o histórico das arquiteturas de CPUs e GPUs, é essencial compreender como esses componentes evoluíram para atender às crescentes demandas de desempenho e eficiência. Além disso, é importante explorar as aplicações atuais dessas arquiteturas, identificando como elas impactam a forma como interagimos com a tecnologia em diversos campos, como jogos, ciência, medicina, inteligência artificial e muito mais.

Neste documento, serão discutidos os principais marcos no desenvolvimento das arquiteturas de CPUs e GPUs, desde suas origens até as inovações mais recentes. Serão apresentadas as características fundamentais dessas arquiteturas, destacando as diferenças entre elas e como elas são aplicadas em diferentes contextos.

2 Visão geral sobre CPUs

A CPU é a Unidade Central de Processamento de um computador que executa operações aritméticas e lógicas com latência mínima [1]. As CPUs executam seu processamento de forma sequencial [2].

A estrutura básica de uma CPU é dividida entre três partes principais [3]:

- **Unidade Lógica e Aritmética (ULA ou ALU):** executa as quatro operações básicas (adição, subtração, multiplicação e divisão) e operações lógicas de Álgebra Booleana, como IF, AND e OR.
- **Unidade de Controle (UC):** tem a responsabilidade de extrair dados da memória, decodificá-los e executá-los, consultando a ULA quando necessário.
- **Registradores:** unidades de memória da CPU, sendo as mais rápidas e caras em sua categoria. Eles são reservados exclusivamente para uso na CPU, dependendo de altas velocidades de acesso.

Em síntese, a CPU é a responsável por processar todas as principais operações de funcionamento de um computador. Por isso, ela é comumente chamada de processador.

Algumas características sobre as CPUs são [2]:

- A CPU desempenha um papel fundamental ao fornecer ao computador um poder de processamento eficiente para executar tarefas gerais diárias com eficiência.
- A CPU executa o processamento serial de tarefas, ou seja, uma tarefa por vez em uma série.
- A CPU possui um número relativamente menor de núcleos em comparação com as GPUs, porém cada núcleo é altamente eficiente e poderoso.

2.1 A História da CPU

As CPUs (Unidades de Processamento Central) têm passado por uma evolução significativa ao longo da história da computação. Desde os primeiros computadores eletrônicos até os sistemas modernos, as CPUs progrediram em termos de desempenho, complexidade e recursos. As primeiras CPUs eram simples e só podiam executar funções aritméticas básicas, enquanto as CPUs de hoje são muito mais avançadas e podem lidar com processamento de dados complexos e multitarefa. As primeiras CPUs em funcionamento surgiram na década de 1970 com o Intel 4004, o primeiro microprocessador do mundo. No entanto, embora a Intel tenha sido uma das primeiras empresas a desenvolver CPUs, concorrentes como AMD e Qualcomm também fizeram contribuições significativas para o campo. [4].

Aqui estão algumas das CPUs mais relevantes na história da computação::

- **Intel 4004 (1971):** Considerado o primeiro microprocessador comercialmente disponível, o Intel 4004 foi um marco na indústria de CPUs. Era um processador de 4 bits com uma velocidade de clock de 740 kHz e foi usado em calculadoras e outros dispositivos eletrônicos.
- **Intel 8086 (1978):** O Intel 8086 foi o primeiro processador da arquitetura x86 e marcou o início da família de processadores Intel 80x86. Ele foi usado nos primeiros computadores pessoais, como o IBM PC original.
- **Motorola 68000 (1979):** O Motorola 68000 foi um processador de 16/32 bits usado em muitos computadores pessoais e estações de trabalho da década de 1980. Sua arquitetura foi considerada avançada na época, com recursos como endereçamento de memória de 32 bits e barramento de dados de 16 bits.
- **Intel Pentium (1993):** O processador Intel Pentium foi um marco significativo na história das CPUs. Introduziu a arquitetura superscalar, que permitia a execução de múltiplas instruções por ciclo de clock. O Pentium foi amplamente utilizado em computadores pessoais durante a década de 1990.
- **AMD Athlon (1999):** O AMD Athlon foi uma linha de processadores que competiu diretamente com os processadores Intel Pentium na época. Os Athlons ofereciam um desempenho notável e trouxeram inovações, como a arquitetura de núcleo Thunderbird e a tecnologia de fabricação em processo de 0,18 mícron.
- **Intel Core i7 (2008):** Os processadores Intel Core i7 introduziram a microarquitetura Nehalem, que trouxe melhorias significativas no desempenho e eficiência energética. Eles foram os primeiros a apresentar a tecnologia Hyper-Threading, que permitia a execução simultânea de várias threads em um único núcleo físico.
- **AMD Ryzen (2017):** A linha de processadores AMD Ryzen marcou um retorno competitivo da AMD no mercado de CPUs. Eles apresentaram a microarquitetura Zen, que oferecia desempenho notável e eficiência energética, além de suporte a um maior número de núcleos e threads em comparação com seus concorrentes.

2.2 Tipos de Processadores

Existem alguns tipos bastante conhecidos de processadores modernos, dentre os mais utilizados estão [2]:

- **Intel Core:** está entre as CPUs mais utilizadas no mercado, os processadores da série Core da Intel são frequentemente encontrados em desktops e laptops.
- **AMD Ryzen:** oferecem forte desempenho a preços baixos e destinam-se a competir com os processadores da série Core da Intel.
- **Processadores ARM:** populares em dispositivos móveis como smartphones e tablets e são conhecidos por serem energeticamente eficientes.

Dois dos principais fabricantes de CPU do mercado hoje são Intel e AMD.

3 Arquitetura de GPUs

A sigla GPU se dá pela abreviação do termo Unidades de Processamento Gráfico, elas foram criadas com o objetivo principal de processar dados de vídeo em alta performance. Por sua alta capacidade de processar dados em paralelos, as GPUs começaram a ser usada em várias outras áreas onde a programação paralela se sobressai, como por exemplo, a extração de criptomoedas e aprendizado de máquina.

As principais características das GPUs são: [1]:

- **Quantidade alta de ULAs:** As GPUs possuem uma quantidade muito elevada de Unidades Lógicas Aritméticas, a fim de gerenciar quantias enormes de dados de forma paralelas.
- **Conectividade através de portas:** Várias portas conseguem ser usadas para conectar os dados da GPU em outros dispositivos, por exemplo, HDMI, VGA e DVI.
- **Aptidão para programação paralela:** São desenvolvidas primariamente para atividades que podem ser feitas de forma paralela.
- **Habilidade de fazer cálculos com ponto flutuante:** As GPUs são capazes de fazer operações com vetores ou números com pontos flutuantes com maior facilidade.

3.1 Comparação entre arquiteturas de CPUs e GPUs

A arquitetura da GPU (Unidade de Processamento Gráfico) difere significativamente da arquitetura da CPU (Unidade de Processamento Central) em vários aspectos-chave. A seguir, apresentamos uma breve comparação entre as duas arquiteturas, destacando suas principais diferenças [1]:

- **Paralelismo Massivo**: A GPU é altamente otimizada para processamento paralelo, com um grande número de núcleos de processamento, enquanto a CPU geralmente possui menos núcleos, sendo mais adequada para tarefas sequenciais.
- **Estrutura de Barramento**: A GPU possui uma estrutura de barramento mais ampla, o que permite a transferência de dados de maneira mais rápida em comparação com a CPU.
- **Frequência de Clock**: A CPU normalmente opera em uma frequência de clock mais alta do que a GPU, o que a torna mais eficiente em tarefas sequenciais e com menor paralelismo.
- **Cache**: A CPU geralmente possui mais níveis de cache e uma hierarquia de cache mais complexa do que a GPU, o que melhora o desempenho em tarefas com menor paralelismo e alta dependência de dados.

- **Arquitetura de Pipeline**: A GPU apresenta uma arquitetura de pipeline mais profunda, permitindo um maior número de instruções sendo processadas simultaneamente. A CPU possui um pipeline mais curto, adequado para um menor número de instruções.
- **Precisão de Ponto Flutuante**: A GPU geralmente possui suporte a uma precisão de ponto flutuante inferior em comparação com a CPU. Essa diferença é aceitável em muitas aplicações gráficas, mas pode afetar a precisão em cálculos científicos intensivos.
- **Arquitetura de Memória**: A GPU possui uma arquitetura de memória mais eficiente para acessos aleatórios e acesso simultâneo de várias threads, enquanto a CPU tem uma arquitetura de memória mais flexível para tarefas com menor paralelismo.
- **Capacidade de Expansão**: A CPU é mais versátil em termos de capacidade de expansão e suporte a diferentes periféricos, como dispositivos de armazenamento e placas de rede. A GPU, por sua vez, é otimizada principalmente para processamento gráfico.
- **Consumo de Energia**: Devido à sua alta quantidade de núcleos de processamento e frequências de clock mais baixas, a GPU geralmente consome mais energia do que a CPU, tornando-a mais adequada para aplicações que requerem grande poder de processamento.
- **Arquitetura SIMD (Single Instruction, Multiple Data)**: A GPU utiliza uma arquitetura SIMD, na qual uma instrução é executada simultaneamente em vários dados, enquanto a CPU emprega uma arquitetura mais flexível, executando instruções de forma independente.
- **Flexibilidade x Eficiência**: A CPU é altamente flexível e pode executar uma ampla gama de tarefas, enquanto a GPU é altamente eficiente para processamento paralelo massivo, mas com menos flexibilidade em relação a tarefas sequenciais.

3.2 Evolução das arquiteturas

As arquiteturas de GPUs têm passado por uma evolução notável ao longo dos anos, impulsionada pela demanda por maior poder de processamento gráfico em jogos, computação científica, aprendizado de máquina e outras aplicações intensivas em computação visual.

Aqui estão alguns exemplos de arquiteturas de GPUs e sua evolução:

- **NVIDIA GeForce 256 (1999):** A GeForce 256 foi uma das primeiras GPUs da NVIDIA a introduzir o conceito de processamento gráfico acelerado por hardware. Ela trouxe recursos avançados para renderização 3D em tempo real e a tecnologia Transform and Lighting (TL), que descarregava tarefas de processamento da CPU para a GPU.
- **ATI Radeon R300 (2002):** A arquitetura R300 introduziu o conceito de shaders programáveis. Isso permitia que os desenvolvedores criassem e executem programas personalizados na GPU para controlar a aparência e o comportamento dos objetos em 3D, resultando em efeitos visuais mais avançados e realistas.

- **NVIDIA GeForce 8 Series (2006):** A arquitetura GeForce 8 Series, também conhecida como GeForce Tesla, trouxe melhorias significativas na eficiência e desempenho energético, além de suporte a recursos avançados, como o Unified Shader Model. Isso permitia a programação flexível de shaders e o uso simultâneo de múltiplos e diferentes tipos de shaders na mesma tarefa.
- **AMD Radeon HD 5000 Series (2009):** A arquitetura Radeon HD 5000 Series, conhecida como Evergreen, foi a primeira a introduzir a tecnologia de processamento em 40 nm. Ela trouxe suporte a DirectX 11 e recursos avançados, como Tessellation, melhorando a qualidade gráfica e a fidelidade visual nos jogos e aplicativos.
- **NVIDIA Turing (2018):** A arquitetura Turing trouxe uma série de avanços significativos na GPU, incluindo o uso de núcleos de processamento Tensor para acelerar tarefas de aprendizado de máquina, a tecnologia Ray Tracing em tempo real para renderização de gráficos mais realistas e o DLSS (Deep Learning Super Sampling) para melhorar o desempenho e a qualidade visual dos jogos.
- **AMD RDNA 2 (2020):** A arquitetura RDNA 2, presente nas GPUs da série AMD Radeon RX 6000, trouxe melhorias significativas no desempenho por watt, além de suporte a recursos como Ray Tracing e Variable Rate Shading. Essa arquitetura foi projetada para fornecer um desempenho excepcional em jogos 3D e aceleração de computação.

É importante destacar que a evolução continua, com o lançamento de novas arquiteturas que trazem aprimoramentos em desempenho, eficiência energética e recursos, buscando atender às demandas cada vez mais exigentes dos aplicativos gráficos e de computação visual.

4 Modelo de programação para GPUs

4.1 CUDA

Em 2006, a NVIDIA lançou a CUDA (Compute Unified Device Architecture), uma plataforma de computação paralela e modelo de programação. Essa plataforma oferece a capacidade de aumentar significativamente o desempenho computacional, aproveitando a potência das GPUs (unidades de processamento gráfico) para processar dados. [5]

O modelo de programação CUDA é baseado em linguagens como C, C++ e Fortran, e requer uma placa de vídeo NVIDIA para aproveitar dessa linguagens. É altamente recomendado utilizar o CUDA quando uma parte do processamento pode ser subdividida e executada em paralelo em vários núcleos. Isso ocorre porque os núcleos da GPU são mais numerosos, embora menos poderosos individualmente. [5]

O modelo de programação CUDA proporciona abstrações simples em relação à organização hierárquica de threads, memória e sincronização, permitindo que os programadores escrevam programas escaláveis sem a necessidade de aprender uma infinidade de novos componentes de programação. A arquitetura CUDA é compatível com várias linguagens e ambientes de programação e tem sido amplamente adotada em aplicações e pesquisas publicadas. Atualmente, ela está presente em notebooks, estações de trabalho, clusters de computação e supercomputadores. [6]

4.2 OpenCL

O OpenCL (Open Computing Language) é uma estrutura projetada para permitir a escrita de programas que podem ser executados em plataformas heterogêneas. Essas plataformas podem incluir uma variedade de dispositivos de processamento, como CPUs, GPUs, DSPs, FPGAs e outros aceleradores de hardware. O OpenCL oferece uma abordagem unificada para o desenvolvimento de software, permitindo que os programadores aproveitem eficientemente o poder de processamento desses diferentes dispositivos em um ambiente de programação comum. Isso possibilita a criação de aplicações que podem ser executadas em uma ampla gama de sistemas com diferentes configurações de hardware.

OpenCL é um padrão aberto e livre de royalties para programação paralela em plataformas diversas, abrangendo uma ampla variedade de aceleradores encontrados em supercomputadores, servidores em nuvem, computadores pessoais, dispositivos móveis e sistemas embarcados. Com o uso do OpenCL, é possível obter melhorias significativas em velocidade e capacidade de resposta em uma ampla gama de aplicativos, abrangendo diversas áreas de mercado, como ferramentas criativas profissionais, software científico e médico, processamento de visão e treinamento e inferência de redes neurais. [7]

Referências Bibliográficas

[1] C. BasuMallick, "Cpu vs. gpu: 11 key comparisons." https://www.spiceworks.com/tech/hardware/articles/cpu-vs-gpu/, 2023. Accessed: 2023-06-11. 3, 5

[2] D. Rana, "Cpu vs gpu: Why gpus are more suited for deep learning?." https://www.analyticsvidhya.com/blog/2023/03/cpu-vs-gpu/, 2023. Accessed: 2023-06-11. 3, 4

[3] R. Gogoni, "O que é cpu?." https://tecnoblog.net/responde/o-que-e-cpu/. Accessed: 2023-06-11. 3

[4] D. Beren, "The history of the modern cpu." https://history-computer.com/the-history-of-the-modern-cpu/, 2023. Accessed: 2023-06-11. 3

[5] "Introdução ao cuda." https://edisciplinas.usp.br/pluginfile.php/4146828/mod_resource/content/1/MaterialCUDA.pdf#:~:text=CUDA%20(Compute%20Unified%20Device%20Architecture,%E2%80%8Bprocessar%E2%80%8B%E2%80%8Bdados. Accessed: 2023-06-11. 8

[6] L. Zanotto, A. Ferreira, and M. Matsumoto, "Arquitetura e programação de gpu nvidia," 2019. Accessed: 2023-06-11. 8

[7] "Khronos." https://www.khronos.org/opencl/. Accessed: 2023-06-11. 8