

Uso de FPGAs para HPC

Harre Ayma, Estelamaris Reyes

IME-USP

July 5, 2023

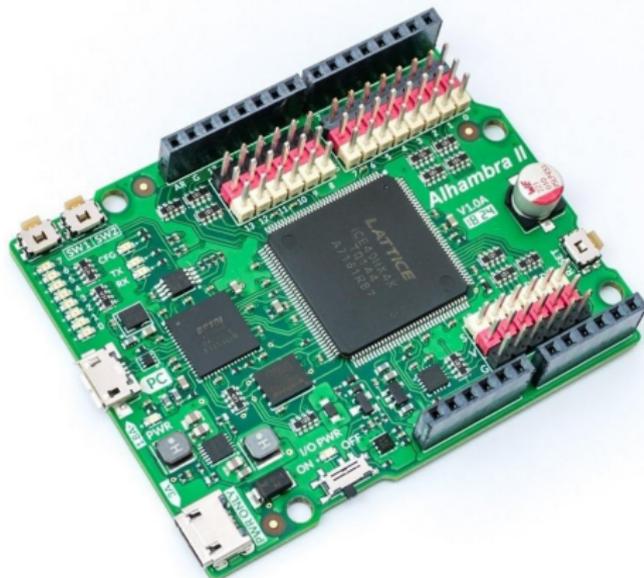
Introdução a FPGA

Introdução a FPGA

- As FPGAs (Field-Programmable Gate Arrays) são dispositivos eletrônicos programáveis que contêm uma matriz de blocos lógicos e conexões configuráveis.

Introdução a FPGA

- As FPGAs (Field-Programmable Gate Arrays) são dispositivos eletrônicos programáveis que contêm uma matriz de blocos lógicos e conexões configuráveis.
- Permitem a implementação de circuitos digitais personalizados por meio da programação da interconexão de seus blocos lógicos.



FPGA: Vantagens

FPGA: Vantagens

- **Reconfigurabilidade:** Uma das principais vantagens das FPGAs é sua capacidade de serem reprogramadas e reconfiguradas após sua fabricação.

FPGA: Vantagens

- **Reconfigurabilidade:** Uma das principais vantagens das FPGAs é sua capacidade de serem reprogramadas e reconfiguradas após sua fabricação.
- **Paralelismo massivo:** As FPGAs são especialmente eficientes na execução de tarefas paralelas devido à sua arquitetura altamente paralela.

FPGA: Vantagens

- **Reconfigurabilidade:** Uma das principais vantagens das FPGAs é sua capacidade de serem reprogramadas e reconfiguradas após sua fabricação.
- **Paralelismo massivo:** As FPGAs são especialmente eficientes na execução de tarefas paralelas devido à sua arquitetura altamente paralela.
- **Desempenho e eficiência energética:** Devido à sua capacidade de paralelismo massivo e ao seu design altamente otimizado, as FPGAs oferecem um desempenho notável em comparação com outros dispositivos.

FPGA: Vantagens

- **Reconfigurabilidade:** Uma das principais vantagens das FPGAs é sua capacidade de serem reprogramadas e reconfiguradas após sua fabricação.
- **Paralelismo massivo:** As FPGAs são especialmente eficientes na execução de tarefas paralelas devido à sua arquitetura altamente paralela.
- **Desempenho e eficiência energética:** Devido à sua capacidade de paralelismo massivo e ao seu design altamente otimizado, as FPGAs oferecem um desempenho notável em comparação com outros dispositivos.
- **Flexibilidade e adaptabilidade:** As FPGAs são altamente flexíveis e adaptáveis a uma ampla gama de aplicações.

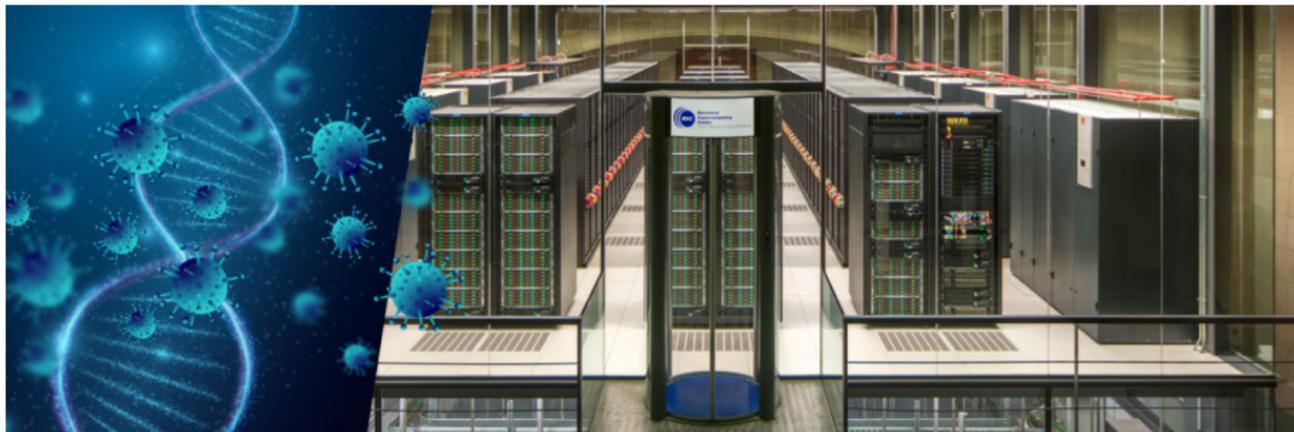
FPGA: Vantagens

- **Reconfigurabilidade:** Uma das principais vantagens das FPGAs é sua capacidade de serem reprogramadas e reconfiguradas após sua fabricação.
- **Paralelismo massivo:** As FPGAs são especialmente eficientes na execução de tarefas paralelas devido à sua arquitetura altamente paralela.
- **Desempenho e eficiência energética:** Devido à sua capacidade de paralelismo massivo e ao seu design altamente otimizado, as FPGAs oferecem um desempenho notável em comparação com outros dispositivos.
- **Flexibilidade e adaptabilidade:** As FPGAs são altamente flexíveis e adaptáveis a uma ampla gama de aplicações.
- **Ciclos de desenvolvimento rápidos:** Graças à sua capacidade de reconfiguração e prototipagem rápida, as FPGAs permitem acelerar os ciclos de desenvolvimento de hardware.

- Processamento de Sinais: As FPGAs são amplamente utilizadas em aplicações de processamento de sinais.

FPGA: Aplicações

- Processamento de Sinais: As FPGAs são amplamente utilizadas em aplicações de processamento de sinais.
- No campo do HPC, as FPGAs podem ser usadas para acelerar algoritmos intensivos em computação e aplicações de processamento de dados em larga escala.



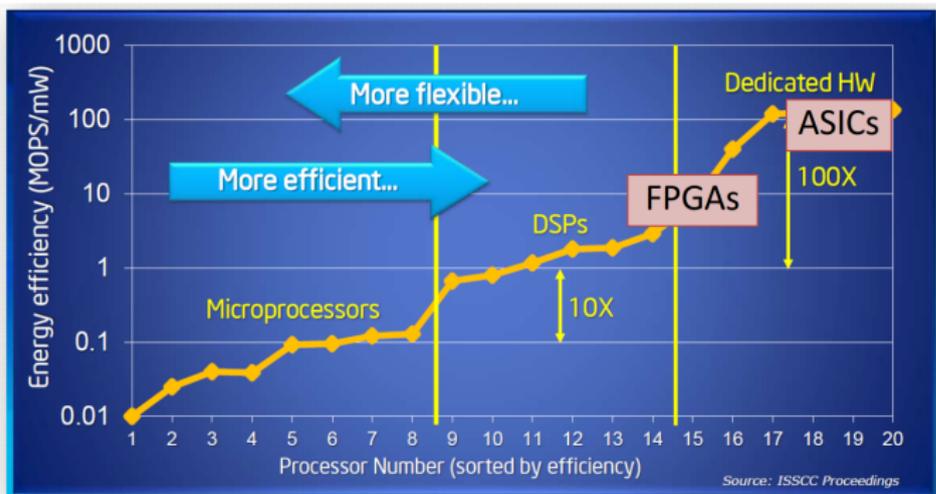
Por que usar FPGAs para HPC?

Por que usar FPGAs para HPC?

- **Alto desempenho e eficiência energética.** As FPGAs são projetadas para realizar operações em paralelo e podem ser adaptadas especificamente aos algoritmos e tarefas computacionais requeridas, proporcionando alto desempenho e eficiência energética.

Por que usar FPGAs para HPC?

- **Alto desempenho e eficiência energética.** As FPGAs são projetadas para realizar operações em paralelo e podem ser adaptadas especificamente aos algoritmos e tarefas computacionais requeridas, proporcionando alto desempenho e eficiência energética.



* Source: Ning Zhang and Bob Brodersen, ISSCC data

Por que usar FPGAs para HPC?

- **Adaptabilidade a diferentes algoritmos.** Uma das vantagens-chave das FPGAs é sua capacidade de reconfiguração em campo, permitindo que sejam adaptadas a diferentes algoritmos.

Por que usar FPGAs para HPC?

- **Adaptabilidade a diferentes algoritmos.** Uma das vantagens-chave das FPGAs é sua capacidade de reconfiguração em campo, permitindo que sejam adaptadas a diferentes algoritmos.
- **Paralelismo e capacidade de processamento massivo.** As FPGAs são altamente paralelas e podem executar múltiplas operações simultaneamente

Casos de uso de FPGAs em HPC

- **Simulação e modelagem computacional.** As FPGAs são valiosas na simulação e modelagem computacional, pois podem acelerar os cálculos intensivos necessários nessas áreas.

- **Simulação e modelagem computacional.** As FPGAs são valiosas na simulação e modelagem computacional, pois podem acelerar os cálculos intensivos necessários nessas áreas.
- **Criptografia e segurança da informação.** As FPGAs oferecem um alto nível de paralelismo e capacidade de processamento, o que as torna adequadas para aplicações de criptografia e segurança da informação.

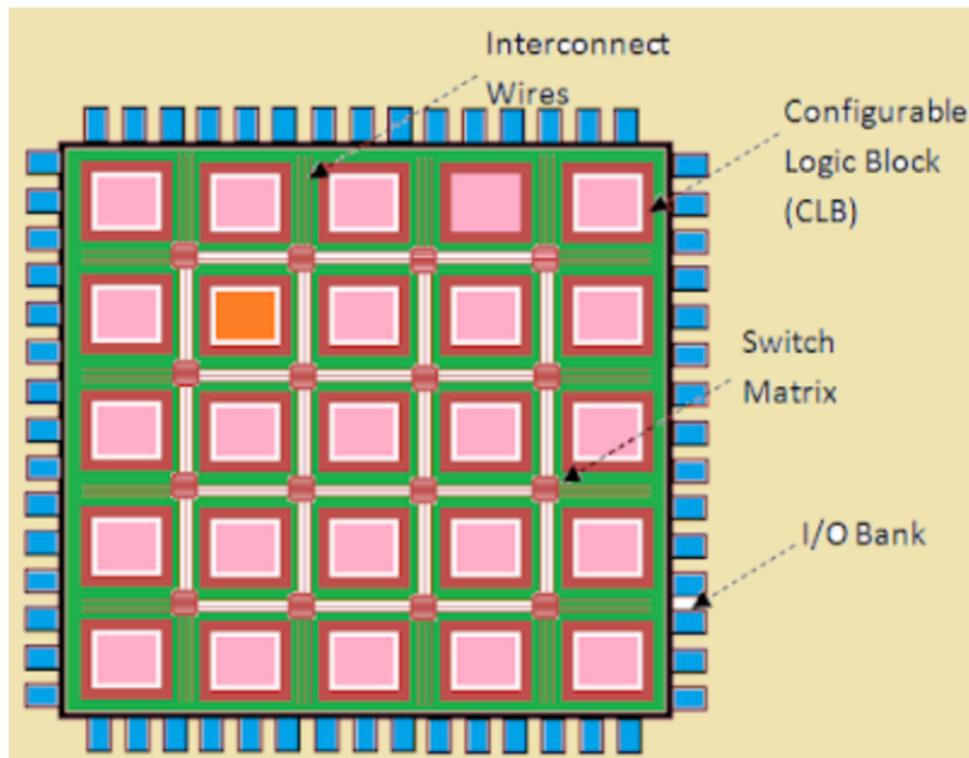
- **Processamento de imagens e visão computacional.** Elas podem acelerar algoritmos de processamento de imagens, detecção de objetos, reconhecimento facial e análise de vídeo em tempo real.

- **Processamento de imagens e visão computacional.** Elas podem acelerar algoritmos de processamento de imagens, detecção de objetos, reconhecimento facial e análise de vídeo em tempo real.
- **Aprendizado de máquina (Machine Learning) e redes neurais.** As FPGAs oferecem uma plataforma de hardware altamente paralela e personalizável que se adapta bem às demandas dos algoritmos de aprendizado de máquina e redes neurais.

- **Processamento de imagens e visão computacional.** Elas podem acelerar algoritmos de processamento de imagens, detecção de objetos, reconhecimento facial e análise de vídeo em tempo real.
- **Aprendizado de máquina (Machine Learning) e redes neurais.** As FPGAs oferecem uma plataforma de hardware altamente paralela e personalizável que se adapta bem às demandas dos algoritmos de aprendizado de máquina e redes neurais.
- **Sequenciamento de DNA em bioinformática.** As FPGAs podem ser programadas e adaptadas especificamente para implementar algoritmos de sequenciamento de DNA, permitindo uma maior otimização e personalização para atender aos requisitos de desempenho e precisão.

Arquitetura de FPGAs para HPC

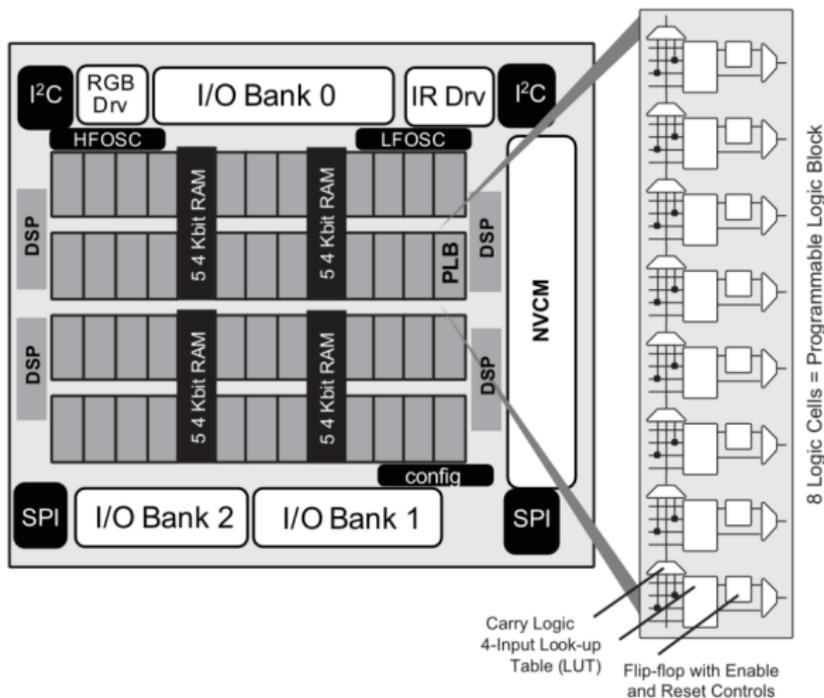
- Arquitetura interna de uma FPGA.



- **CLB(Configurable Logic Block) o PLB:** É uma unidade de hardware programável que consiste em uma matriz de elementos lógicos, como portas lógicas, multiplexadores e flip-flops, interconectados por uma rede de interconexão programável. O CLB permite a implementação de circuitos lógicos personalizados ao configurar a conexão e funcionalidade de seus elementos internos. Os CLBs podem ser reprogramados para se adequar a diferentes algoritmos e aplicações.
- **Matriz de comutação(Switch Matrix),** Ela atua como um conjunto de comutadores eletrônicos controlados que podem ser configurados para estabelecer as conexões necessárias entre os blocos de lógica e os recursos de entrada/saída da FPGA. Essa matriz de comutação proporciona flexibilidade e permite a configuração de rotas de sinal personalizadas, permitindo a adaptação da FPGA a diferentes projetos e aplicações.

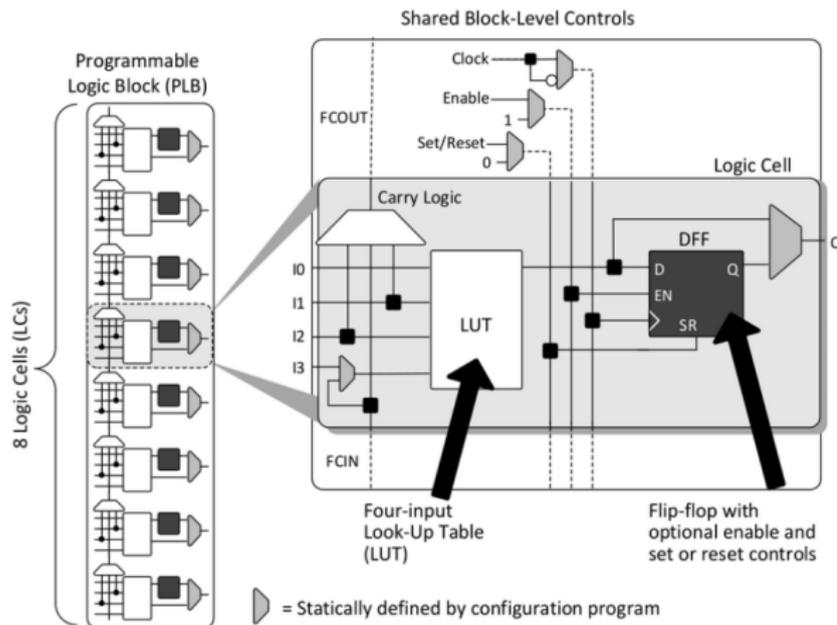
Arquitetura de FPGAs para HPC

- Descrição da arquitetura interna de uma FPGA



Arquitectura de FPGAs para HPC

- Configurable Logic Block (CLB o PLB)

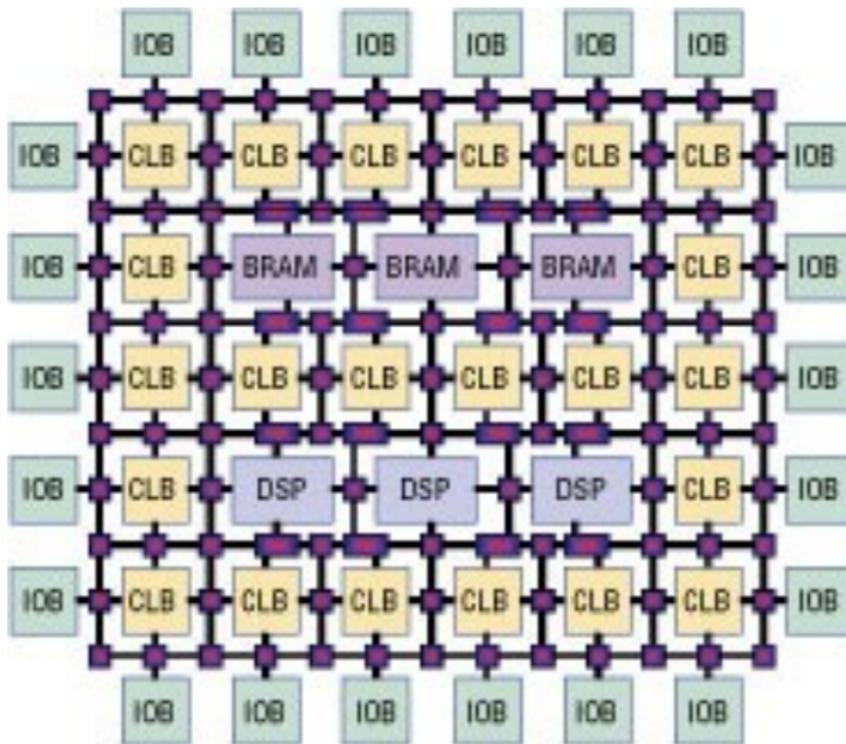


11 -Programmable Logic Block (PLB) of the Lattice Semiconductor ICE40 UltraPlus FPGA family. Source: reproduced from LATTICE SEMICONDUCTOR (2018).

- **Blocos de RAM (memória de acesso aleatório):** Eles fornecem armazenamento temporário de dados e permitem acesso rápido às informações. Esses blocos são usados para armazenar e recuperar dados em aplicações que exigem acesso eficiente a grandes volumes de informações.
- **Blocos DSP (Processamento Digital de Sinais):** São recursos especializados que oferecem capacidade de processamento de sinais em tempo real. Esses blocos contêm unidades aritméticas dedicadas e circuitos otimizados para realizar operações matemáticas complexas de maneira eficiente. Eles são usados em aplicações de processamento de sinais, como processamento de áudio, vídeo, telecomunicações e processamento de imagens.
- **PLL (Phase-Locked Loop):** É um circuito que gera um sinal de clock com uma frequência estável e controlada, o que permite sincronizar o funcionamento dos diferentes elementos da FPGA

Arquitectura de FPGAs para HPC

- Arquitectura de FPGAs



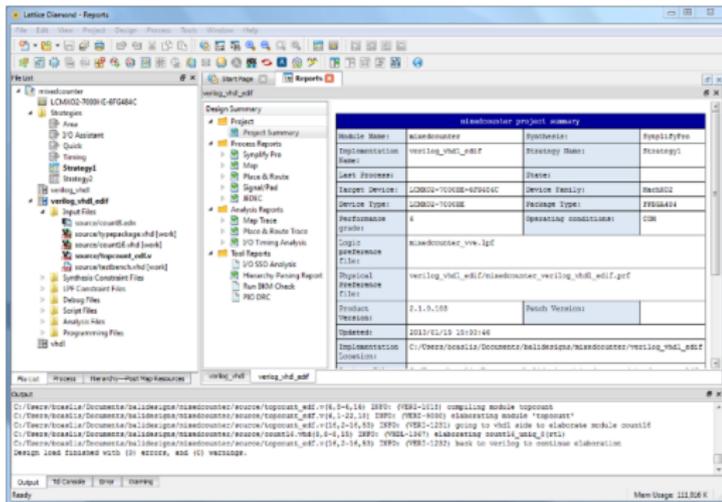
Arquitetura de FPGAs para HPC

- Os **CLBs** podem ser programados para implementar algoritmos paralelos e acelerar operações intensivas de processamento de dados.
- Os blocos de memória (**BRAM**) podem ser usados para armazenar grandes conjuntos de dados, permitindo acesso rápido e eficiente durante o processamento.
- Os **DSPs** (Processadores Digitais de Sinais) oferecem capacidades de processamento de sinais em tempo real e são amplamente utilizados em algoritmos intensivos em matemática e processamento de dados, proporcionando maior desempenho e eficiência energética em aplicações de HPC.

As FPGAs podem ser configuradas e programadas especificamente para se adaptarem aos algoritmos. Isso permite uma otimização eficiente dos recursos, como o uso de memória e a alocação de operações nos blocos lógicos da FPGA. Como resultado, é alcançado um uso eficiente dos recursos e uma melhoria no desempenho geral.

Ferramentas de desenvolvimento de FPGAs

- As ferramentas de design de FPGA mais populares.
 - Xilinx Vivado: Uma das ferramentas mais amplamente utilizadas para projetar e programar FPGAs da Xilinx.
 - Intel Quartus Prime: Uma suíte de ferramentas fornecida pela Intel para projetar FPGAs da família Intel FPGA.
 - Lattice Diamond: Uma ferramenta de design utilizada para desenvolver projetos com FPGAs da Lattice Semiconductor



Ferramentas de desenvolvimento de FPGAs

- Exemplos de linguagens de descrição de hardware (HDL) como Verilog e VHDL.
 - Verilog: Uma linguagem de descrição de hardware popular usada para modelar e projetar circuitos digitais em FPGAs.
 - VHDL (VHSIC Hardware Description Language): Uma linguagem de descrição de hardware amplamente utilizada para projetar sistemas digitais complexos em FPGAs.

```
Library IEEE;
USE IEEE.Std_logic_1164.all;

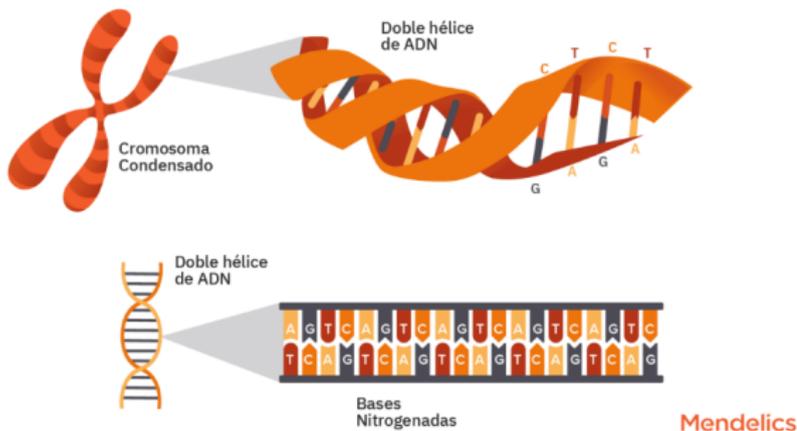
entity RisingEdge_DFlipFlop_SyncReset is
  port(
    Q      : out std_logic;
    Clk    : in  std_logic;
    sync_reset : in  std_logic;
    D      : in  std_logic
  );
end RisingEdge_DFlipFlop_SyncReset;

architecture Behavioral of RisingEdge_DFlipFlop_SyncReset is
begin
  process(Clk)
  begin
    if (rising_edge(Clk)) then
      if (sync_reset='1') then
        Q <= '0';
      else
        Q <= D;
      end if;
    end if;
  end process;
end Behavioral;
```

- **Complexidade do design e programação:** A utilização de FPGAs requer conhecimentos especializados em design de hardware e programação em linguagens de descrição de hardware, o que pode ser complexo e exigir habilidades adicionais.
- **Escalabilidade e custo de implementação:** A adaptação de aplicações de HPC para FPGAs requer um planejamento cuidadoso para garantir a escalabilidade do sistema e pode envolver custos significativos de desenvolvimento e implementação.
- **Ciclo de desenvolvimento mais longo em comparação com soluções baseadas apenas em software:** A implementação de algoritmos em FPGAs pode demandar mais tempo devido à necessidade de otimização de hardware e testes adicionais, o que pode prolongar o ciclo de desenvolvimento do projeto.

Estudo de caso: Sequenciamento de DNA e sua relação com HPC

- O sequenciamento de DNA é o processo de determinar a sequência exata de nucleotídeos em uma molécula de DNA.



- Os avanços no sequenciamento de DNA permitem gerar grandes volumes de dados em pouco tempo. Porém, a análise e processamento desses dados representam um desafio computacional.

A solução está na Computação de Alto Desempenho (HPC).

Algoritmo Smith-Waterman

O algoritmo de Smith-Waterman é um algoritmo de programação dinâmica usado para realizar o alinhamento local de sequências de DNA ou proteínas. O algoritmo consiste em dois passos principais:

- **Cálculo da matriz de pontuação de similaridade:** Uma matriz de pontuação de similaridade é criada, representando a similaridade entre cada par de caracteres das duas sequências.
- **Busca da subsequência ótima:** A busca pela subsequência mais similar entre as duas sequências é feita usando a matriz de pontuação de similaridade.

```
Função SmithWaterman(s1, s2, match, mismatch, gap):  
    n = comprimento(s1);  
    m = comprimento(s2);  
    matriz = matriz de tamanho  $(n + 1) \times (m + 1)$  inicializada com 0;  
    pontuação_max = 0;  
    posição_max = (0, 0);  
    for i = 1 até n do  
        for j = 1 até m do  
            if s1[i] == s2[j] then  
                pontuação = match;  
            else  
                pontuação = mismatch;  
            end
```

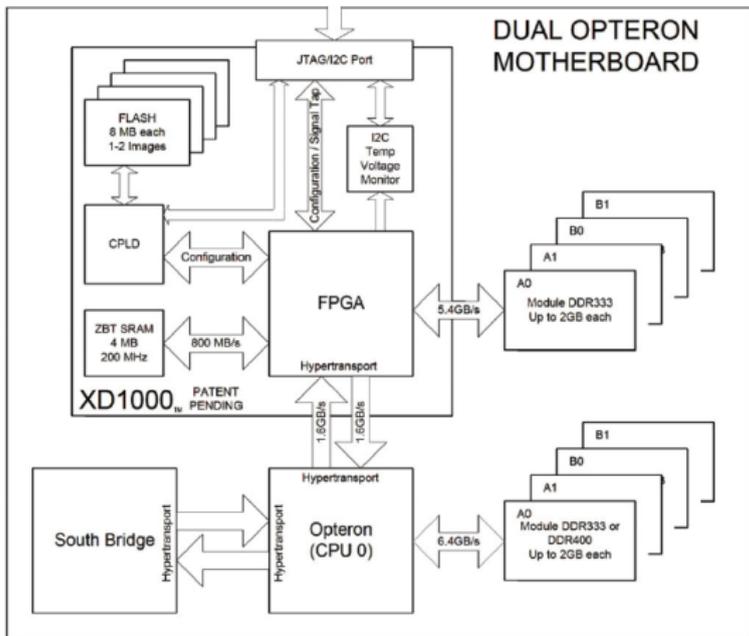
Implementação do Algoritmo Smith-Waterman em uma plataforma de supercomputação reconfigurável

Devido à grande quantidade de dados que precisam ser processados nas comparações de sequências de DNA, o algoritmo de Smith-Waterman pode ser implementado em plataformas de supercomputação e aceleradores de hardware como FPGA para melhorar o desempenho e a eficiência do processamento.

- Zhang et al.[1] apresentaram a implementação do algoritmo de Smith-Waterman na plataforma de supercomputação reconfigurável XD1000 da XtremeData Inc. Seu objetivo foi demonstrar como a plataforma FPGA pode melhorar significativamente o desempenho e a eficiência do processamento do algoritmo de Smith-Waterman para aplicações de sequenciamento de DNA e proteínas.

Implementação do Algoritmo Smith-Waterman em uma plataforma de supercomputação reconfigurável

A FPGA acelera o processamento do algoritmo de Smith-Waterman ao permitir a implementação de múltiplas unidades de processamento em paralelo.



Resultados de desempenho e eficiência

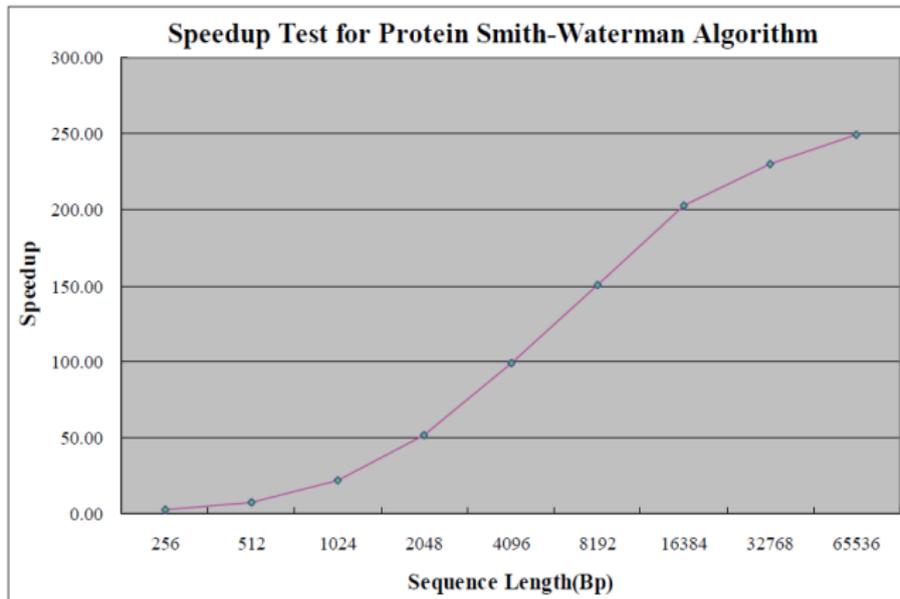


Figure: Os resultados dos testes de velocidade (speedup) do algoritmo Smith-Waterman implementado em um FPGA com 384 PE (Processing Elements) em comparação com uma implementação em uma CPU convencional (Central Processing Unit)

- O potencial e as vantagens de usar FPGAs em aplicações de HPC.
 - As FPGAs possibilitam aceleradores personalizados para cada aplicação, aumentando o desempenho e reduzindo o consumo de energia. A latência também é menor em comparação com soluções baseadas em CPUs/GPUs convencionais.
 - As FPGAs são altamente flexíveis, permitindo atualizações rápidas e adaptações em comparação com hardwares fixos. Sua adaptabilidade torna as FPGAs ideais para lidar com tarefas de processamento intensivo em aplicações de HPC.
 - O co-design de software e hardware permite a otimização conjunta dos algoritmos e da arquitetura FPGA, adaptando o hardware aos requisitos específicos da aplicação e otimizando o desempenho em tempo real, como no caso do sequenciamento de DNA.

-  P. Zhang, G. Tan, and G. R. Gao, “Implementation of the smith-waterman algorithm on a reconfigurable supercomputing platform,” in *Proceedings of the 1st international workshop on High-performance reconfigurable computing technology and applications: held in conjunction with SC07, 2007*, 2007, pp. 39–48.
-  M. Taheri, H. Zandevakili, and A. Mahani, “A high-performance memristor-based smith-waterman dna sequence alignment using fpni structure,” *Journal of Applied Research in Electrical Engineering*, vol. 1, no. 1, pp. 59–68, 2022.
-  M. Alser, H. Hassan, A. Kumar, O. Mutlu, and C. Alkan, “Shouji: a fast and efficient pre-alignment filter for sequence alignment,” *Bioinformatics*, vol. 35, no. 21, pp. 4255–4263, 2019.