RESEARCH ARTICLE ECEJOURNALS.IN

Developing FPGA-Based Accelerators for Deep Learning in Reconfigurable Computing Systems

Dr T M Sathish Kumar

Associate Professor, Department of Electronics and Communication Engineering, KSR College of Engineering

KEYWORDS:
FPGA accelerators,
Deep learning,
Reconfigurable computing,
Hardware optimization.

ARTICLE HISTORY:
Submitted: 14.01.2024
Revised: 16.02.2024
Accepted: 24.02.2024

DOI:
https://doi.org/10.31838/rcc/01.01.01

Abstract

The incorporation of Field-Programmable Gate Arrays (FPGAs) into deep learning frameworks has paved the way for significant improvements in computational performance and energy efficiency within reconfigurable computing systems. This study investigates the creation and deployment of FPGA-based accelerators designed specifically for deep learning tasks. It begins with a comprehensive overview of the architectural design principles and hardware aspects pertinent to FPGA accelerators. The analysis then shifts to performance metrics, evaluating FPGA accelerators against conventional GPU and CPU systems in terms of speed, efficiency, and scalability. Furthermore, the paper explores various optimization strategies aimed at enhancing energy efficiency and throughput in FPGA implementations. Practical applications and advantages of FPGA accelerators are highlighted through case studies in real-world deep learning contexts. The study concludes with a discussion on future trends and challenges, underlining the potential of FPGAs to foster innovation in deep learning and reconfigurable computing. This research underscores the pivotal role of FPGAs in elevating the capabilities of deep learning systems, providing detailed insights into their development and optimization.

Author's e-mail: tmsathish123@gmail.com

How to cite this article: Sathish Kumar M T, Developing FPGA-Based Accelerators for Deep Learning in Reconfigurable Computing Systems. SCCTS Transactions on Reconfigurable Computing, Vol. 1, No. 1, 2024 (pp. 1-5).

Introduction

The fusion of Field-Programmable Gate Arrays (FPGA) with deep learning has gained substantial attention recently, as it offers the potential to enhance computational tasks while maintaining flexibility and efficiency. FPGA-based accelerators present a promising solution to the increasing demands of deep learning applications, especially in reconfigurable computing systems. Figure 1 shows the architecture of FPGA [1]. This introduction delves into the importance of FPGAs in deep learning, the benefits they offer, and the challenges and future prospects in this evolving field.

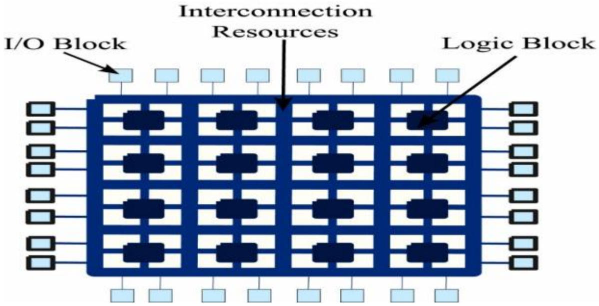

Figure 1. FPGA architecture

Deep learning, a branch of machine learning, involves neural networks with multiple layers that can learn and make intelligent decisions from extensive data sets. These networks require significant computational power for both training and inference, which has traditionally been provided by Central Processing Units (CPUs) and Graphics Processing Units (GPUs) [2]. While GPUs have become the standard for deep learning tasks due to their parallel processing capabilities, they have limitations in power consumption, latency, and flexibility. FPGAs address these limitations by offering a reconfigurable hardware solution that can be tailored to specific computational needs.

FPGAs are integrated circuits that can be programmed and reprogrammed to perform a variety of tasks. Unlike fixed-function devices, FPGAs allow designers to create custom hardware configurations optimized for specific applications. This adaptability is particularly advantageous in deep learning, where different models and architectures may require unique hardware configurations for optimal performance. Utilizing FPGAs, developers can design accelerators that provide high throughput, low latency, and energy-efficient computation tailored to the needs of deep learning tasks. Framework of FPGA-based hardware accelerator is shown in Figure 2 [3].

One of the primary advantages of FPGA-based accelerators is their ability to provide customized parallelism. Deep learning tasks often involve matrix multiplications and other operations that can be parallelized. FPGAs can be configured to exploit this parallelism, resulting in significant performance gains [4]. For instance, an FPGA can be programmed to perform multiple operations simultaneously, reducing the time required for computation compared to sequential processing in CPUs or less efficient parallelism in GPUs. This capability is particularly beneficial for real-time applications, such as autonomous driving or medical imaging, where rapid processing of large datasets is crucial.

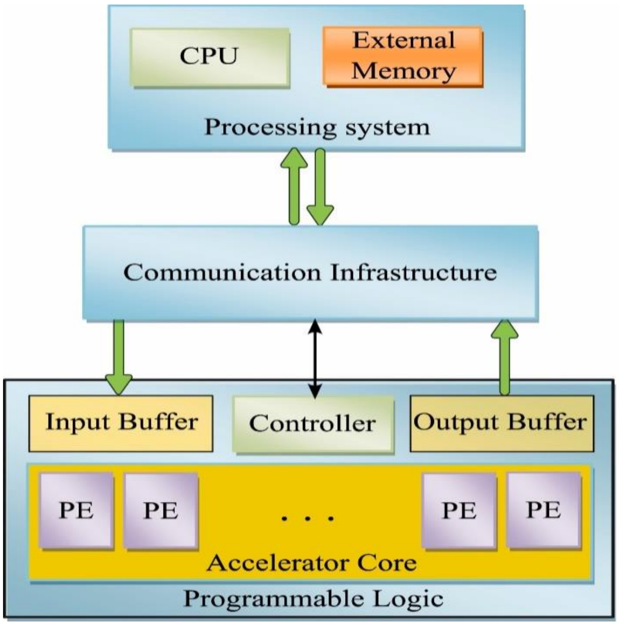

Figure 2. framework of FPGA-based hardware accelerator

Moreover, FPGAs offer superior energy efficiency compared to traditional processors. As deep learning models grow in complexity and size, the power consumption of GPUs becomes a critical concern [5]. FPGAs, with their custom hardware configurations, can achieve the same or better performance with significantly lower power consumption. This efficiency is essential for deploying deep learning models in edge devices and environments where power resources are limited. For example, in IoT devices or remote sensing applications, FPGAs can provide the computational power needed without draining battery life.

Despite these advantages, there are challenges associated with FPGA-based accelerators for deep learning [6]. One of the primary challenges is the complexity of FPGA programming. Designing efficient hardware configurations requires expertise in hardware description languages (HDLs) such as VHDL or Verilog, which can be a barrier for software developers accustomed to high-level programming languages. To address this, recent advancements in high-level synthesis (HLS) tools have been made, allowing developers to design FPGA configurations using more familiar languages like C/C++. These tools automatically translate high-level code into HDL, simplifying the design process and making FPGA programming more accessible.

In conclusion, FPGA-based accelerators represent a powerful and flexible solution for the demanding computational needs of deep learning. Their ability to provide customized parallelism, energy efficiency, and reconfigurability makes them well-suited for a wide range of applications, from real-time processing to edge computing. While challenges remain, ongoing advancements in FPGA technology and design tools are poised to overcome these hurdles, paving the way for more widespread adoption and innovation in the field. As the landscape of deep learning continues to evolve, FPGAs will undoubtedly play a crucial role in shaping the future of reconfigurable computing systems.

Architecture and Design Principles

The architecture and design of FPGA-based accelerators for deep learning are pivotal in maximizing the potential of Field-Programmable Gate Arrays (FPGAs) for computational tasks [7]. Unlike conventional processors, FPGAs offer a unique blend of reconfigurability and parallelism, enabling customized hardware configurations tailored to specific deep learning models and applications.

Central to FPGA-based accelerators is their architecture optimized for parallel processing. FPGAs consist of an array of configurable logic blocks (CLBs) interconnected by programmable routing channels. These CLBs can be configured to implement intricate digital circuits, making FPGAs highly adaptable to a variety of computational tasks. In the context of deep learning, the architecture typically includes specialized components such as multipliers, adders, and memory blocks, which can be interconnected and customized to efficiently execute neural network computations.

Designing FPGA-based deep learning accelerators involves mapping the computational tasks of neural networks onto the FPGA architecture. This process includes dividing the neural network into layers and assigning each layer to appropriate FPGA resources. For example, matrix multiplication, a fundamental operation in deep learning, can be accelerated by mapping it onto parallel processing units within the FPGA. Additionally, optimizing data movement between FPGA components and external memory is critical to minimize latency and maximize throughput.

The architecture also integrates memory hierarchy optimizations to manage data efficiently. FPGAs typically include on-chip memory blocks (Block RAM) used to store weights, activations, and intermediate results. By reducing the need to access off-chip memory, these on-chip resources enhance performance and reduce energy consumption. Furthermore, FPGA-based accelerators often incorporate high-bandwidth interfaces such as PCIe or high-speed memory interfaces (e.g., DDR) to facilitate efficient data exchange with host systems or external storage.

Design principles for FPGA-based deep learning accelerators emphasize achieving a balance between computation, memory access, and interconnect efficiency. Techniques such as pipelining, exploiting parallelism, and sharing resources are employed to maximize hardware utilization and throughput. Moreover, optimizing the placement and routing of logic within the FPGA fabric is crucial to minimize critical path delays and ensure consistent performance.

Performance Evaluation of FPGA Accelerators

Evaluating the performance of FPGA accelerators in deep learning involves examining their computational efficiency, throughput, latency, and energy consumption relative to traditional processors and GPU-based solutions [8]. FPGAs offer unique advantages such as reconfigurability and parallelism, which significantly impact their performance metrics.

Throughput, a crucial measure, gauges the speed at which computations are processed. FPGAs achieve high throughput by exploiting parallelism across their configurable logic blocks (CLBs) and specialized processing units designed for neural network tasks. Optimizations in data flow, pipelining strategies, and efficient memory access patterns are employed to maximize computational efficiency and throughput.

Latency, another critical metric, measures the time taken to complete a single inference or training operation. FPGA accelerators minimize latency by leveraging their parallel processing capabilities, allowing simultaneous execution of multiple operations. Techniques like algorithmic pipelining and resource sharing within the FPGA fabric help reduce latency, enhancing responsiveness for applications requiring real-time performance.

Energy efficiency is a key consideration, with FPGAs known for their low power consumption compared to GPUs and CPUs. Evaluating energy efficiency involves measuring power usage during operation and assessing how effectively the FPGA utilizes resources to perform neural network computations.

Benchmarking plays a pivotal role in performance evaluation, providing an objective basis for comparing FPGA accelerators against other hardware platforms. Standard benchmarks in deep learning tasks, such as ImageNet classification or neural network training scenarios, are executed on FPGA-based systems. Performance metrics like frames per second (FPS), operations per second (OPS), and power efficiency metrics such as performance per watt are used to quantify and compare FPGA performance across different configurations and workloads.

Energy Efficiency and Optimization Techniques in FPGA-Based Deep Learning

Enhancing energy efficiency in FPGA-based deep learning systems is crucial given the rising computational requirements and the imperative to minimize power consumption. FPGAs offer distinct advantages like reconfigurability and parallel processing, which can be optimized to improve energy efficiency in deep learning tasks [9].

An effective strategy to optimize energy use in FPGA-based deep learning accelerators involves architectural design. This entails tailoring the FPGA architecture to meet the specific demands of neural network models. Techniques such as pruning redundant connections, reducing computational precision through lower-bit fixed-point arithmetic, and optimizing memory access patterns help reduce power consumption without sacrificing accuracy. By aligning the FPGA architecture with the neural network's structure and computational requirements, significant improvements in energy efficiency can be achieved.

Algorithmic optimizations also play a critical role in minimizing energy consumption. Approaches like algorithmic pipelining, where multiple computation stages overlap to maximize resource utilization, and efficient scheduling of operations within the FPGA fabric, can reduce idle cycles and lower power consumption. These optimizations ensure that FPGA resources are utilized efficiently during inference and training tasks, thereby enhancing overall energy efficiency.

Effective power management strategies further contribute to energy optimization in FPGA-based deep learning accelerators. Dynamic voltage and frequency scaling (DVFS) techniques adjust the operating voltage and frequency of FPGA components based on workload demands, optimizing power usage while maintaining performance levels. Additionally, techniques like clock gating and power gating selectively disable unused components or reduce their clock frequency during periods of inactivity to conserve power without compromising functionality.

Hardware-software co-design methodologies also play a significant role in achieving energy efficiency. By partitioning computational tasks between FPGA hardware and software (CPU or GPU), workloads can be optimized to leverage the strengths of each platform. This approach minimizes power consumption by offloading intensive computations to the FPGA hardware while efficiently managing less demanding tasks on the CPU or GPU.

Case Studies: Implementation of FPGA Accelerators in Real-World Deep Learning Applications

Implementing FPGA accelerators in real-world deep learning applications has demonstrated significant advantages, primarily in enhancing performance and efficiency compared to traditional computing architectures. One notable case study involves the use of FPGAs for accelerating convolutional neural networks (CNNs), which are foundational in image and video processing tasks.

In the realm of computer vision, FPGA-based accelerators have been deployed to accelerate tasks such as object detection, facial recognition, and image segmentation. These applications benefit from the parallel processing capabilities of FPGAs, where customizable hardware implementations of CNN layers can achieve high throughput and low latency [10]. For instance, researchers and developers have integrated FPGA-based accelerators into surveillance systems, autonomous vehicles, and medical imaging devices, where real-time processing and low power consumption are critical requirements.

Another compelling case study involves natural language processing (NLP) applications, such as language translation and sentiment analysis. FPGAs offer advantages in optimizing the performance of recurrent neural networks (RNNs) and transformer-based models like BERT (Bidirectional Encoder Representations from Transformers) [11]. These models require extensive computation for tasks such as sequence prediction and language understanding. By offloading compute-intensive tasks to FPGA accelerators, these applications achieve faster inference times and reduced energy consumption, making them suitable for deployment in resource-constrained environments.

Furthermore, FPGA accelerators have shown promise in accelerating reinforcement learning algorithms used in robotics and gaming. These algorithms rely on iterative processes that benefit from the parallelism and low-latency characteristics of FPGAs. For example, FPGA-based implementations have been explored in training robotic agents for complex tasks such as navigation, manipulation, and decision-making in dynamic environments.

Despite these successes, implementing FPGA accelerators in real-world applications poses challenges related to hardware design complexity, programming paradigms, and integration with existing software frameworks. Addressing these challenges requires expertise in FPGA development, optimization techniques tailored to specific deep learning models, and seamless integration into the overall system architecture.

Future Directions and Challenges in FPGA-Based Deep Learning Accelerators

Looking forward, the future trajectory of FPGA-based accelerators in deep learning presents both promising avenues and persistent challenges. One key direction involves refining FPGA architectures and design methodologies to effectively support the evolving landscape of deep learning models. As neural networks grow in complexity and diversity, FPGA designs must adapt to efficiently handle larger models with increasingly demanding computational requirements.

Additionally, there is a strong emphasis on enhancing the flexibility and programmability of FPGA-based accelerators. This involves exploring new programming frameworks and development tools tailored to simplify the process of deploying deep learning applications on FPGA platforms. Such advancements aim to lower barriers to adoption and foster innovation in this specialized domain.

Energy efficiency remains a critical hurdle for FPGA-based accelerators. Despite their inherent advantages in power consumption over CPUs and GPUs, further optimizing energy efficiency is essential for scaling these accelerators in environments with strict energy constraints and portable devices. Techniques such as dynamic voltage and frequency scaling (DVFS), power gating, and algorithmic optimizations specific to FPGA architectures will play pivotal roles in achieving these efficiency goals.

Looking ahead, there is also growing interest in exploring heterogeneous computing architectures that integrate FPGAs with other accelerators like GPUs and ASICs. Such hybrid architectures can leverage the unique strengths of each technology to achieve superior performance and efficiency for specific deep learning tasks, such as training large-scale models or conducting real-time inference at the edge.

Addressing these future directions involves tackling challenges such as hardware complexity, high development costs, and the specialized expertise required for FPGA programming and optimization. Collaboration between academic research and industry will be crucial to advancing FPGA technology, standardizing development tools, and establishing best practices for seamlessly integrating FPGA-based accelerators into mainstream deep learning workflows. These efforts are poised to unlock the full potential of FPGA accelerators across diverse applications in artificial intelligence and machine learning.

REFERENCES

[1] Altman, Morteza Babaee, et al. "Machine learning algorithms for FPGA Implementation in biomedical engineering applications: A review." Heliyon (2024).

[2] Zhang, Chen, et al. "Optimizing FPGA-based accelerator design for deep convolutional neural networks." Proceedings of the 2015 ACM/SIGDA international symposium on field-programmable gate arrays. 2015.

[3] Altman, Morteza Babaee, et al. "Machine learning algorithms for FPGA Implementation in biomedical engineering applications: A review." Heliyon (2024).

[4] Chen, Yiran, et al. "A survey of accelerator architectures for deep neural networks." Engineering 6.3 (2020): 264-274.

[5] George, Varghese, and Jan M. Rabaey. Low-Energy FPGAs—Architecture and Design. Vol. 625. Springer Science & Business Media, 2012.

[6] Wang, Teng, et al. "An overview of FPGA based deep learning accelerators: challenges and opportunities." 2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS). IEEE, 2019.

[7] Shawahna, Ahmad, Sadiq M. Sait, and Aiman El-Maleh. "FPGA-based accelerators of deep learning networks for learning and classification: A review." ieee Access 7 (2018): 7823-7859.

[8] Nguyen, Tan, et al. "FPGA-based HPC accelerators: An evaluation on performance and energy efficiency." Concurrency and Computation: Practice and Experience 34.20 (2022): e6570.

[9] Feng, Gan, et al. "Energy-efficient and high-throughput FPGA-based accelerator for Convolutional Neural Networks." 2016 13th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT). IEEE, 2016.

[10] Feng, Xin, et al. "Computer vision algorithms and hardware implementations: A survey." Integration 69 (2019): 309-320.

[11] Wang, Chenghao, and Zhongqiang Luo. "A review of the optimal design of neural networks based on FPGA." Applied Sciences 12.21 (2022): 10771.