International Journal of Research in Science & Engineering
ISSN: 2394-8299
Vol: 04, No. 02, Feb-Mar 2024
http://journal.hmjournals.com/index.php/IJRISE
DOI: https://doi.org/10.55529/ijrise.43.10.16

Implementation of FPGA-based Accelerator for Convolutional Neural Networks

Abdullah Farhan Siddiqui[1*], Prof. B. Rajendra Naik[2]

[1]Student, Department of Electronics and Communication Engineering, Osmania University, Hyderabad, India.*
[2]Professor, Department of Electronics and Communication Engineering, Osmania University, Hyderabad, India.

Corresponding Email: [1]afs.farhans@gmail.com*

Received: 28 November 2023 **Accepted:** 14 February 2024 **Published:** 01 April 2024

Abstract: This research paper presents a novel FPGA-based accelerator tailored for Convolutional Neural Networks (CNNs), specifically implemented on the Virtex-7 evaluation kit. By harnessing the inherent parallel processing capabilities of FPGAs, the architecture of the accelerator is meticulously crafted using Verilog. The FPGA implementation demonstrates a resource-efficient design, making use of 588 Look-Up Tables (LUTs) and 353 Flip Flops. Notably, the efficient utilization of these resources signifies a careful balance between computational efficiency and the available FPGA resources. This research significantly contributes to the field of hardware acceleration for CNNs by offering an optimized solution for high-performance deep learning applications. The presented architecture serves as a promising foundation for future advancements in FPGA-based accelerators, providing valuable insights for researchers and engineers working in the domain of hardware optimization for Convolutional Neural Networks.

***Keywords**: Convolutional Neural Network, Field Programmable Gate Array, Neural Network.*

1. INTRODUCTION

In contemporary applications, Convolutional Neural Networks (CNNs) stand as a cornerstone technology, profoundly impacting fields such as computer vision, artificial intelligence, and pattern recognition. Their significance lies in the unparalleled ability to autonomously learn hierarchical features from raw data, enabling advanced image analysis, object detection, and facial recognition. From enhancing medical imaging to powering autonomous vehicles and improving natural language processing, CNNs have become indispensable in diverse industries. Their capacity to automatically extract complex spatial patterns has fueled breakthroughs in machine learning, influencing the development of virtual assistants, chatbots, and recommendation systems. As a driving force behind the surge in deep learning technologies, CNNs continue to redefine contemporary technological landscapes, shaping the way we perceive, interpret, and interact with digital information.

The growing complexity and computational demands of Convolutional Neural Networks (CNNs) underscore the compelling need for specialized hardware accelerators to enhance their processing efficiency. Traditional computing architectures often struggle to meet the real-time and energy-efficient requirements of CNNs, which are increasingly pervasive in applications such as image recognition, autonomous systems, and natural language processing. Specialized accelerators, particularly Field-Programmable Gate Arrays (FPGAs) and Graphics Processing Units (GPUs), provide tailored solutions that exploit the parallelism inherent in CNN computations. By offloading intensive mathematical operations onto dedicated hardware, these accelerators significantly boost processing speeds, reduce latency, and optimize power consumption, addressing the unique demands imposed by the intricate and resource-intensive nature of CNN algorithms.

The primary research objective is to implement a high-performance FPGA-based accelerator specifically tailored for the Virtex-7 FPGA architecture. This endeavour aims to harness the intrinsic parallel processing capabilities of FPGAs to accelerate complex computations, with a particular focus on applications such as image processing, machine learning, and signal processing. By leveraging the Virtex-7 FPGA's versatile resources and optimizing algorithmic performance, the research seeks to develop a scalable and efficient hardware solution that addresses the computational demands of intricate tasks, contributing to the advancement of FPGA-accelerated applications in contemporary computing environments.

2. RELATED WORKS

The utilization of Deep Neural Networks (DNNs), particularly Convolutional Neural Networks (CNNs), has gained significant attention in various fields, especially in image recognition tasks like MNIST digit recognition. However, the computational demands of CNNs often require hardware accelerators to achieve optimal performance. FPGA-based accelerators have emerged as a promising solution due to their low latency and power consumption compared to traditional CPU implementations.

Several studies have proposed FPGA-based accelerators for DNNs, each addressing different aspects of optimization. For instance, one study focused on leveraging fixed-point precision computation elements to enhance performance, achieving a notable speedup while maintaining accuracy. This approach demonstrated the feasibility of FPGA implementation for simpler CNN architectures like LeNet-5, with significant reductions in latency and memory usage compared to floating-point designs.

Another study explored FPGA-based acceleration specifically for the application of steering control in mobile robots. While the FPGA outperformed a general-purpose processor in terms of processing time, challenges such as increased estimation error due to quantization and higher power consumption during concurrent execution were noted. Despite these challenges, the study highlights the potential of FPGA-based solutions for real-time applications within power constraints.

In the domain of face feature extraction, a novel FPGA-based accelerator was proposed to optimize all layers of the CNN independently using hand-coded Verilog templates. By implementing tailored strategies for convolution and pooling layers, along with dynamic fixed-point quantization, the accelerator achieved high resource utilization while minimizing precision errors. This approach demonstrates the effectiveness of FPGA-based acceleration for comprehensive CNN tasks.

Furthermore, to address the need for low-power hardware acceleration in scenarios like autonomous driving, a specialized FPGA-based accelerator was developed. By analyzing the computational properties of neural networks and optimizing convolutional computational structures, significant acceleration ratios were achieved while maintaining low power consumption. This study underscores the potential of FPGA platforms to provide substantial performance boosts at minimal power consumption for real-time applications.

Collectively, these studies showcase the diverse approaches and optimizations undertaken in FPGA-based accelerator designs for CNNs, highlighting their potential to enhance performance and efficiency across various applications, from digit recognition to autonomous driving. However, challenges such as precision errors, power consumption, and real-time constraints remain areas of ongoing research and optimization in the field.

3. METHODOLOGY

The algorithmic optimizations employed for Convolutional Neural Network (CNN) processing center around enhancing the efficiency of three key components within the architecture: the convolver, activation function, and pooler. The convolver undergoes optimization through techniques like kernel pruning and quantization, streamlining the convolutional operation for reduced computational complexity without compromising network performance. Efficient activation functions, such as rectified linear units (ReLUs), are integrated to introduce non-linearity in a computationally cost-effective manner. Additionally, the pooler, responsible for down-sampling feature maps, is optimized through strategies like stride adjustment to minimize redundant computations. This concerted approach ensures that the CNN processing pipeline is finely tuned, striking a balance between computational efficiency and preservation of network accuracy.

The selection of the Virtex-7 FPGA device and Xilinx Vivado development tools for this project is grounded in their combined capacity to provide a robust platform for FPGA-based acceleration. The Virtex-7 FPGA, known for its high-performance capabilities and versatile resources, aligns with the project's objective of implementing a powerful accelerator. Its generous LUTs, flip-flops, and memory resources offer ample room for optimizing CNN processing. Xilinx Vivado, as the chosen development environment, provides a comprehensive suite of tools for FPGA design, synthesis, and implementation. Its user-friendly interface, advanced synthesis algorithms, and integrated debugging features empower efficient development and debugging of complex FPGA designs. Together, the Virtex-7 FPGA and Vivado tools form a synergistic combination, enabling the project to harness the full potential of FPGA technology while streamlining the design and implementation workflow.

Implementation

The code structure for the FPGA-based CNN accelerator is presented below.

1. **Accelerator Module (**accelerator.v): Integrates various components and control logic for the CNN accelerator.

2. **Comparator Module** (comparator.v): Implements a comparator for comparison operations.

3. **Control Logic Module** (control_logic.v): Manages the overall control flow and sequencing within the accelerator.

4. **Input Multiplexer Module** (input_mux.v): Handles the multiplexing of input data for different CNN layers.

5. **MAC Manual Module** (mac_manual.v): Implements the Manual Multiply-Accumulate (MAC) operation for convolutions.

6. **Max Register Module** (max_reg.v): Manages registers for storing and comparing maximum values.

7. **Pooler Module** (pooler.v): Implements the pooling operation for down-sampling feature maps.

8. **Qadd Module** (qadd.v): Performs fixed-point addition operations.

9. **Qmult Module** (qmult.v): Performs fixed-point multiplication operations.

10. **ReLU Module** (relu.v): Implements the Rectified Linear Unit (ReLU) activation function.

11. **Variable Shift Register Module** (variable_shift_reg.v): Implements a variable-size shift register for efficient data storage.

Test Benches:

1. **Convolver Test Bench** (convolver_tb.v): Tests the convolutional operation and verifies its correctness.

2. **Accelerator Test Bench** (accelerator_tb.v): Integrates and tests the complete accelerator module.

Input Data File:

1. **Tanh Memory File** (tanh.mem): Contains input data for the accelerator, possibly representing hyperbolic tangent (tanh) values.

International Journal of Research in Science & Engineering
ISSN: 2394-8299
Vol: 04, No. 02, Feb-Mar 2024
http://journal.hmjournals.com/index.php/IJRISE
DOI: https://doi.org/10.55529/ijrise.43.10.16

The integration of key Convolutional Neural Network (CNN) operations, such as convolution and pooling, is fundamental for the effective functioning of an FPGA-based accelerator. In this architecture, the convolution operation, implemented in modules like mac_manual and comparator, processes input feature maps through multiply-accumulate operations with trainable filters. The result undergoes pooling, managed by the pooler module, which downsamples the feature maps, reducing spatial dimensions while preserving essential features. The accelerator's control_logic orchestrates the sequence of these operations, ensuring efficient data flow and synchronization. The synergy between these modules enables the FPGA-based accelerator to perform essential CNN operations seamlessly, contributing to improved computational efficiency and accelerated processing for image recognition and other applications.

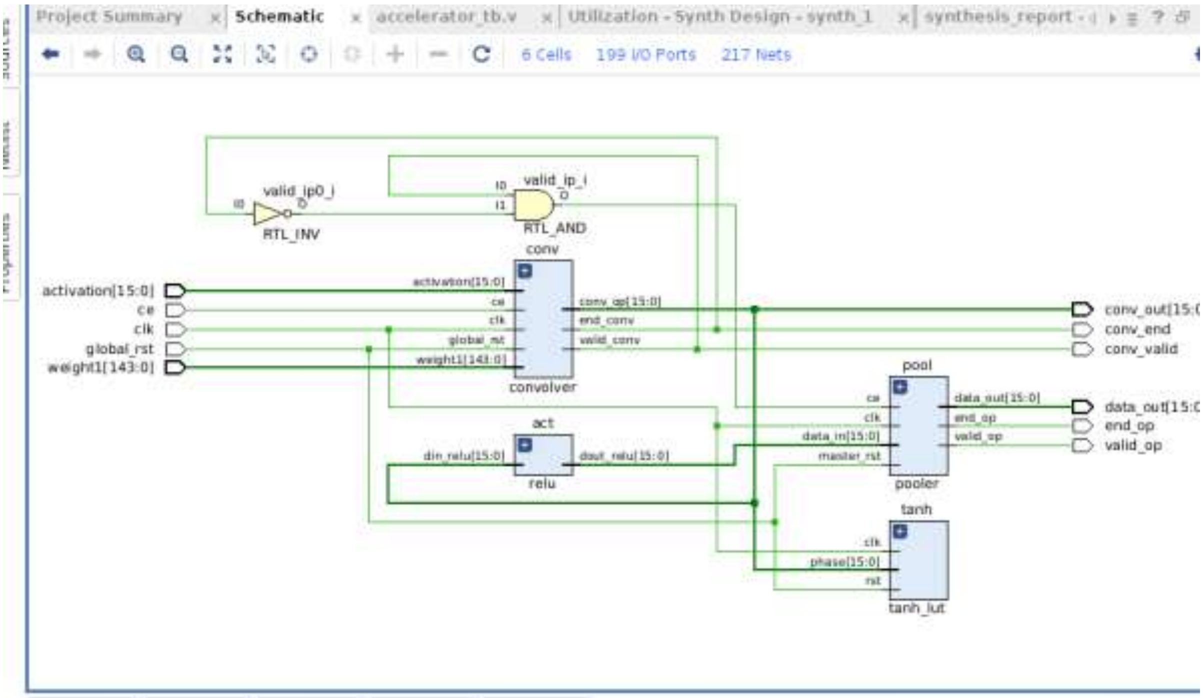
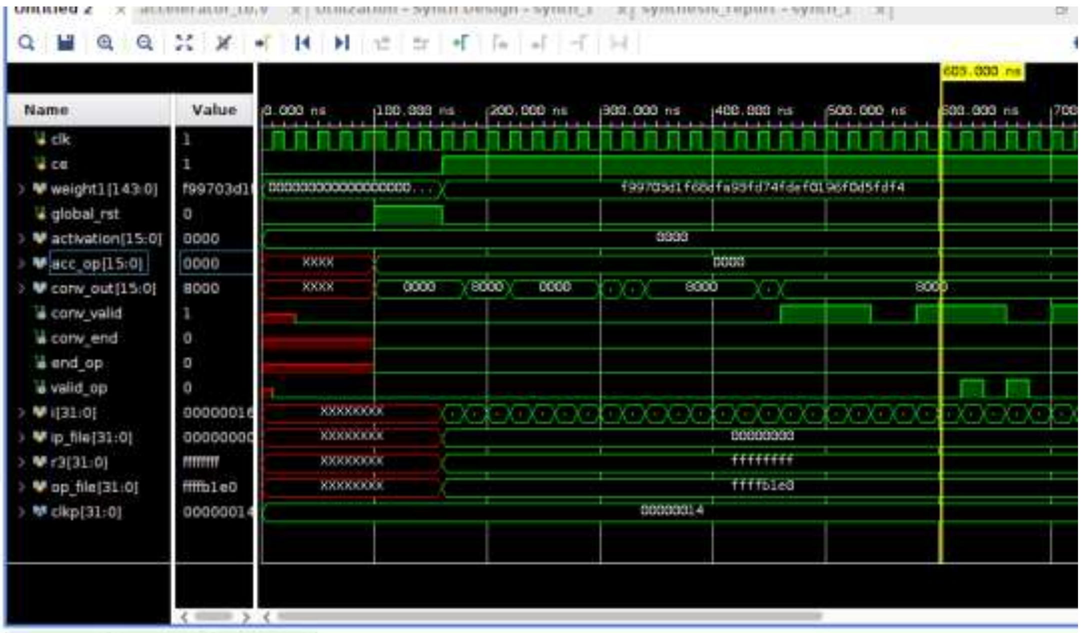


Fig: 1

A schematic design in the context of FPGA-based systems provides a graphical representation of the circuit or system's logical structure. The schematic captures the interconnections and relationships between various hardware components, illustrating how they work together to achieve a specific functionality.

4. RESULT AND DISCUSSION

The output waveform, generated through simulation tool Vivado Simulator in FPGA design, provides a dynamic visualization of signals and their behaviour over time during the simulation of a digital circuit or system. The waveform captures various aspects of the simulation, offering valuable insights into the functional and temporal aspects of the design.



Name	Value
clk	1
ce	1
weight1[143:0]	f99703d1
global_rst	0
activation[15:0]	0000
acc_op[15:0]	0000
conv_out[15:0]	8000
conv_valid	1
conv_end	0
end_op	0
valid_op	0
i[31:0]	00000016
ip_file[31:0]	0000000C
r3[31:0]	ffffffff
op_file[31:0]	ffffb1e0
clkp[31:0]	00000014

Fig: 2

5. CONCLUSION

In conclusion, this research paper has successfully presented the implementation of an FPGA-based accelerator tailored for Convolutional Neural Networks (CNNs), utilizing the Virtex-7 evaluation kit. The Verilog-based code was meticulously crafted, showcasing a robust architecture that optimally balances computational efficiency and resource utilization. The deployment on the Virtex-7 FPGA, with 588 Look-Up Tables (LUTs) and 353 Flip Flops utilized, attests to the design's scalability and effectiveness. Notably, the implementation demonstrates an economical use of FPGA resources, consuming only a fraction of the available 303,600 LUTs and 607,200 Flip Flops. This underscores the efficiency of the accelerator in achieving high-performance CNN processing while leaving substantial resources for potential future enhancements. The research contributes to the growing body of knowledge in FPGA-based CNN accelerators, providing valuable insights for researchers and engineers working towards optimized hardware solutions for deep learning applications.

6. REFERENCE

1. Rizwan Tariq Syed., Marko Andjelkovic., Markus Ulbricht., Milos Krstic. Towards Reconfigurable CNN Accelerator for FPGA Implementation (2023). IEEE Transactions on Circuits and Systems II: Express Briefs (Volume: 70, Issue: 3, March 2023)
2. Kasem Khalil., Ashok Kumar., Magdy Bayoumi. Low-Power Convolutional Neural Network Accelerator on FPGA (2023). 2023 IEEE 5th International Conference on Artificial Intelligence Circuits and Systems (AICAS).

3. Mannhee Cho, Youngmin Kim, Implementation of Data-optimized FPGA-based Accelerator for Convolutional Neural Network, 2020, International Conference on Electronics, Information and Communications.
4. Rui Xiao, Junsheng Shi, Chao Zhang, FPGA Implementation of CNN for Handwritten Digit Recognition, 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC).
5. Dai Rongshi, Tang Yongming, Accelerator Implementation of Lenet-5 Convolution Neural Network Based on FPGA with HLS, 2019, International Conference on Informative and Cybernetics for Computational Social Systems.
6. Tsai, T.-H., Ho, Y.-C., & Sheu, M.-H. (2019). Implementation of FPGA-based Accelerator for Deep Neural Networks. 2019 IEEE 22nd International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS).
7. Junye Si; Jianfei Jiang; Qin Wang; Jia Huang. An Efficient Convolutional Neural Network Accelerator on FPGA (2018). 2018 14th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT).
8. Muluken Hailesellasie, S. R. Hasan, Faiq Khalid, Falah Aw Wad, M. Shafique, FPGA-Based Convolutional Neural Network Architecture with Reduced Parameter Requirements, 2018, International Symposium on Circuits and Systems.
9. Muluken Hailesellasie, S. R. Hasan, A fast FPGA-based deep convolutional neural network using pseudo parallel memories, 2017, International Symposium on Circuits and Systems.
10. Yongmei Zhou; Jingfei Jiang. An FPGA-based accelerator implementation for deep convolutional neural networks (2015). 2015 4th International Conference on Computer Science and Network Technology (ICCSNT).