ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 9, Issue 3, April 2024

Efficient Neural Network Multiplication Techniques for Low-Power Devices

[1] J.Sarasawathi, [2]Dr.B.Hemalatha
[1] PG Student, VLSI Design
[2]Associate Professor, Department Of Electronics And Communication Engineering
Thirumalai Engineering College

Abstract: ***The "Booth Encoding-Based Energy Efficient Multipliers for Deep Learning Systems" project addresses the pressing need for energy-efficient hardware solutions in deep learning. As AI applications become increasingly power-hungry, our project offers an innovative approach to tackle this challenge. By leveraging Booth encoding and Exponent-of-Two (EO2) quantization, we aim to significantly reduce energy consumption in neural network computations without compromising accuracy. This project promises to extend the battery life of portable devices and minimize the power footprint of neural network accelerators, meeting the growing demand for energy-efficient AI hardware solutions. Additionally, it is designed for effective implementation using Xilinx ISE 14.7, making it a practical and accessible solution for FPGA-based deep learning systems.***

I. INTRODUCTION

This research brief proposes a novel re-encoding scheme aimed at reducing the size of deep neural network (DNN) weights, facilitating advancements in artificial intelligence (AI) at the edge. By leveraging Booth encoding and extended power-of-two (EO2) quantization, the scheme enables highly efficient energy computations during neural network inference while maintaining minimal impact on classification accuracy. The effectiveness of the re-encoding approach is demonstrated through the computation of both convolutional neural networks (CNNs) and linear neural networks. Two specific multipliers, the Extended Exact Multiplier and the EO2 Multiplier, are introduced. The EO2 quantization and re-encoding method achieve a 30.77% reduction in model size for CNNs and a 49.86% reduction for linear neural networks.

Additionally, the introduced multipliers contribute to significant reductions in inference energy. Specifically, the EO2 Multiplier reduces inference energy for CNNs by 50.6% and for linear neural networks by 90.1%. For sensor-end computation of the linear neural network, the EO2 Multiplier demonstrates a 77.32% reduction in area compared to an exact Booth multiplier and a 93.2% reduction in inference energy consumption compared to the unmodified exact multiplier. The proposed scheme not only enhances energy efficiency during inference but also allows for minor adjustments to re-encoding signal arrangements. This combination of the proposed re-encoding scheme and multipliers outperforms existing designs in terms of resource utilization while maintaining a minimal impact on neural network inference accuracy. This research is structured in two phases: Phase 1 involves the Extended Exact Multiplier, and Phase 2 focuses on the EO2 Multiplier, with the latter proving to be more efficient.

Drawbacks

- Increased Hardware Complexity
- Potential Latency
- Error Propagation
- Limited Benefit for Small Multipliers

II. Existing System

Booth encoding is a technique utilized in digital circuit design to optimize the execution of multiplication operations. It achieves this by analyzing the binary representation of the multiplier and exploiting patterns within it. Rather than generating a partial product for each individual bit of the multiplier, Booth encoding identifies sequences of adjacent bits with the same value and combines them into larger groups. By doing so, it reduces the overall number of partial products required for the

multiplication process, thereby improving efficiency. For example, in a traditional multiplication operation, each bit of the multiplier triggers the generation of a corresponding partial product. However, with Booth encoding, consecutive bits of the same value are grouped together, allowing for the creation of larger partial products. This grouping effectively reduces the computational workload and resource utilization during multiplication.

Proposed System

In conclusion, this proposed work endeavors to contribute significantly to the domain of energy-efficient hardware for deep learning applications. By marrying the efficiency gains afforded by Booth encoding with the versatile capabilities of Xilinx 14.7, we aspire to deliver multiplier designs that not only outperform existing solutions but also pave the way for sustainable and power-conscious advancements in the broader field of neural network accelerators. Through meticulous exploration, innovation, and validation, this research aims to make tangible strides towards meeting the burgeoning computational demands of contemporary deep learning while prioritizing energy efficiency in hardware implementations.

Advantages

1. The project significantly improves the energy efficiency of multipliers, reducing power consumption during deep learning inference, which is critical for battery-operated and power-constrained devices.
2. Utilizing Booth encoding and exponent-of-two (EO2) quantization, the project achieves a substantial reduction in model size, enabling more efficient storage and deployment of neural networks.
3. Despite the energy-saving techniques, the project maintains minimal loss in classification accuracy, ensuring reliable performance in deep learning applications.
4. The proposed re-encoding scheme can be applied to existing Booth multiplier designs with minor modifications, making it a practical enhancement for a wide range of hardware.
5. The project's design outperforms existing solutions in terms of resource utilization, making it a compelling choice for FPGA-based deep learning systems.

System Study

Introduction To Modelsim

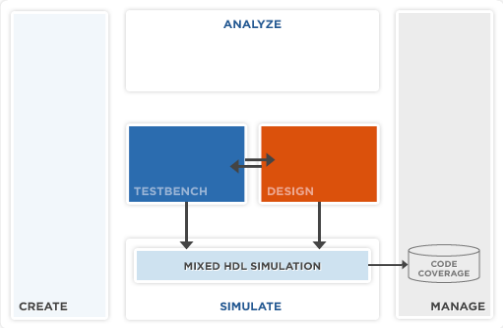
ModelSim is a useful tool that allows you to stimulate the inputs of your modules and view both outputs and internal signals. It allows you to do both behavioral and timing simulation, however, this document will focus on behavioral simulation. Keep in mind that these simulations are based on models and thus the results are only as accurate as the constituent models. ModelSim /VHDL, ModelSim /VLOG, ModelSim /LNL, and ModelSim /PLUS are produced by Model Technology™ Incorporated. Unauthorized copying, duplication, or other reproduction is prohibited without the written consent of Model Technology. The information in this manual is subject to change without notice and does not represent a commitment on the part of Model Technology. The program described in this manual is furnished under a license agreement and may not be used or copied except in accordance with the terms of the agreement. The online documentation provided with this product may be printed by the end-user. The number of copies that may be printed is limited to the number of licenses purchased. ModelSim is a registered trademark of Model Technology Incorporated. Model Technology is a trademark of Mentor Graphics Corporation. PostScript is a registered trademark of Adobe Systems Incorporated. UNIX is a registered trademark of AT&T in the USA and other countries. FLEXlm is a trademark of Globetrotter Software, Inc. IBM, AT, and PC is registered trademarks, AIX and RISC System/6000 are trademarks of International Business Machines Corporation. Windows, Microsoft, and MS-DOS are registered trademarks of Microsoft Corporation. OSF/Motif is a trademark of the Open Software Foundation, Inc. in the USA and other countries. SPARC is a registered trademark and SPARCstation is a trademark of SPARC International, Inc. Sun Microsystems is a registered trademark, and Sun, SunOS and Open Windows are trademarks of Sun Microsystems, Inc. All other trademarks and registered trademarks are the properties of their respective holders.

Standards Supported

ModelSim VHDL supports both the IEEE 1076-1987 and 1076-1993 VHDL, the 1164-1993 Standard Multivalve Logic System for VHDL Interoperability, and the 1076.2-1996 Standard VHDL Mathematical Packages standards. Any design developed with ModelSim will be compatible with any other VHDL system that is compliant with either IEEE Standard 1076-1987 or 1076-1993. ModelSim Verilog is based on IEEE Std 1364-1995 and a partial implementation of 1364-2001, Standard Hardware Description Language Based on the Verilog Hardware Description Language. The Open Verilog International Verilog LRM version 2.0 is also applicable to a large extent. Both PLI (Programming Language Interface) and VCD (Value Change Dump) are supported for ModelSim PE and SE users.

Modelsim - Advanced Simulation And Debug

ASIC and FPGA design



ASIC and FPGA design

Mentor Graphics was the first to combine single kernel simulator (SKS) technology with a unified debug environment for Verilog, VHDL, and System C. The combination of industry-leading, native SKS performance with the best integrated debug and analysis environment make **ModelSim** the simulator of choice for both ASIC and FPGA designs. The best standards and platform support in the industry make it easy to adopt in the majority of process and tool flows.

III. Conclusion

In our extensive study, we aimed to optimize Deep Artificial Neural Networks (DANN) by applying a novel re-encoding process to all fully connected layers, except for the output layer. This strategic approach was implemented in two different neural network architectures, leading to significant reductions in network size. Specifically, the feedforward neural network exhibited a remarkable size reduction of up to 49.8%, while the convolutional neural network (CNN) saw a substantial decrease of up to 30.7%.To further improve the efficiency of our re-encoding strategy, we optimized the Booth multiplier with a base-4 configuration, aligning it with the proposed methodology. The integration of our innovative Extended Exact Multiplier produced significant results, demonstrating a notable reduction in energy consumption during inference for both CNNs and linear neural networks—amounting to 50.6% and 91.1%, respectively.

In parallel with our efforts to optimize multipliers, we introduced two low-performance network devices, referred to as EO2 devices, characterized by minimal latency, power consumption, and overall performance. The integration of the EO2 digital equivalent, along with the enhanced Booth multiplier, resulted in a remarkable 94.2% reduction in power consumption compared to traditional multipliers. Importantly, our proposed multipliers demonstrated superior energy efficiency without sacrificing accuracy, highlighting their potential for widespread application in neural network design. These findings represent a significant advancement towards achieving more resource-efficient and environmentally friendly neural network implementations.

Results

Our innovative re-encoding approach introduces an advanced rounding mechanism that approximates values to the nearest power of two, resulting in a streamlined product chart. This process creates a sawtooth pattern in the absolute error, which decreases as we progress further from the re-coding stage, demonstrating the systematic effectiveness of our strategy. A key component of our approach is the inclusion of a multiplexer (MUX) that intelligently selects outputs based on equivalent values, ensuring numerical balance across layers.

To enhance our methodology, we introduce an improved bit manipulation (EBM) technique, which minimizes information loss during the re-coding process. Integrated exponential-of-two (EO2) multipliers further improve precision, thereby increasing the robustness and efficiency of the neural network. In a dual-layer configuration, one layer implements the original re-coding scheme enhanced with EO2 and transfer multipliers, while the other adheres to the traditional network structure. Each layer undergoes the proposed re-coding process, using original 8-bit quantized weights as approximations for practical hardware implementation.

Our proposed EBM and EO2 multipliers consistently outperform existing methods in both theoretical power efficiency and neural network accuracy. This establishes our re-encoding strategy as a leader in the optimization field, with promising implications for future advancements.

References

1. Abed K.H. and R. E. Siferd, “CMOS VLSI implementation of a low-power logarithmic converter,” IEEE Transactions on Computers, vol. 52, no. 11, pp. 1421–1433, 2003.
2. Al-Qutayri, M., Stouraitis, T.& Alsuhli, G. (2022, October) Tolba, M., Tesfai, H., Saleh, H., Zerom, B.,. Approximate Logarithmic Multiplier For Convolutional Neural Network Inference With Computational Reuse. In 2022 29th IEEE International Conference on Electronics, Circuits and Systems (ICECS) (pp. 1-4). IEEE.
3. Chandra, V. (2017)Lai, L., Suda, N., &. Deep convolutional neural network inference with floating-point weights and fixed-point activations. arXiv preprint arXiv:1703.03073.
4. Del Barrio, A. A.,Kim, Hermida, R., & Bagherzadeh, N. (2018) M. S., , Oliveira, L. T.,. Efficient Mitchell’s approximate log multipliers for convolutional neural networks. IEEE Transactions on Computers, 68(5), 660-675.
5. Gu, F. Y., Lin, C., & Lin, J. W. (2022). A low-power and high-accuracy approximate multiplier with reconfigurable truncation. IEEE Access, 10, 60447-60458.
6. J. Han, J. Liang, and F. Lombardi, “New metrics for the reliability of approximate and probabilistic adders,” IEEE Transactions on computers, vol. 62, no. 9, pp. 1760–1771, 2013.
7. C. Wang and J. C. Principe, “Training neural networks with additive noise in the desired signal,” IEEE Transactions on Neural Networks, vol. 10, no. 6, pp. 1511–1517, 1999.
8. Zhang, H., & Ko, S. B. (2022). Variable-Precision Approximate Floating-Point Multiplier for Efficient Deep Learning Computation. IEEE Transactions on Circuits and Systems II: Express Briefs, 69(5), 2503.
9. Ramdane Haider, M. H., Hina, M.D., Soukane, A., & Ko, S. B. (2023). Booth encoding based energy efficient multipliers for deep learning systems. IEEE Transactions on Circuits and Systems II: Express Briefs.?