|  |  |
| --- | --- |
|  | **2023** |
|  | **ENGI-9819 (Computer Hardware Found)**  Authors: 1. Towhidul Islam – 202381732 (FPGA) 2. Moni Kishore Dhar – 202380330 (ASIC) 3. Sachi Datta – 202387871 (GPU) 4. Rifat Bin Masud – 202387267 (SiLaGO) |

|  |
| --- |
| **[Hardware Architectures for Deep Learning Models]** |
| [Type the abstract of the document here. The abstract is typically a short summary of the contents of the document. Type the abstract of the document here. The abstract is typically a short summary of the contents of the document.] |

****

Table of Contents

[Introduction 3](#_Toc151422217)

[Different types of Hardware Architectures for Deep learning 3](#_Toc151422218)

[i. The components of the architecture: 6](#_Toc151422219)

[Conclusion 6](#_Toc151422220)

[References 7](#_Toc151422221)

**Hardware Architectures for Deep Learning Models**

## Introduction

Deep learning has transformed various fields, from computer vision to natural language processing, by allowing the development of highly precise and sophisticated machine learning models. To meet the computational demands of deep learning, a range of hardware architectures have emerged, each with its own unique robustness and trade-offs. We have studied several papers and found many types of hardware architecture but according to the paper [3] there are four leading hardware architectures for deep learning: Field-Programmable Gate Arrays (FPGAs), Graphics Processing Units (GPUs), Application-Specific Integrated Circuits (ASICs), and the SilaGo method. So, we extensively research those four-hardware architecture according to there their key characteristics, performance, and compatibility for a range of deep learning tasks, offering insights into the evolving landscape of hardware for AI.

## Different types of Hardware Architectures for Deep learning

1. **FPGA:** FPGA stands for Field-Programmable Gate Array. It is a type of integrated circuit that can be programmed and reprogrammed after forming. FPGAs are greatly compatible and can be customized to perform a various task, such as digital signal processing, machine learning, and image processing. They consist of configurable logic blocks and programmable I/O cells that can be programmed to perform specific functions. FPGAs are usually used in applications that need high performance and less power consumption, including deep learning. They are also less expensive to structure and construct than ASICs, allowing them a renown choice for prototyping and less volume output.

**i. DESIGN OF FPGA-BASED DEEP LEARNING ACCELERATORS:** Designing Accelerators for Specific Applications utilizing FPGA-based accelerators for specific problems is currently the most extensive area of applications FPGA accelerator.

Designing an accelerator specifically for a specific problem, it not only fits the problem well but also has a relatively small design difficulty. Designing accelerators

for specific problems often speed up the reasoning process of deep learning algorithms rather than the learning process.

The paper [1] used FPGA to design a dedicated acceleration device to implement the LSTM algorithm to achieve an efficient speech recognition engine (ESE). To speed up predictions and save energy, they used a load-balanced sensing pruning method that compresses the LSTM model size by 20x (10x form pruning, 2x form quantization) with negligible loss of prediction accuracy. The compressed model is then encoded and split into multiple PEs for parallelism, and a complex LSTM data stream is scheduled using a separately designed scheduler. Finally, an ESE hardware architecture that directly runs the sparse LSTM model is implemented. The ESE is implemented in a Xilinx XCKU060 FPGA operating at 200MHz and operates directly on a sparse LSTM network with a performance of 282GOPS, corresponding to a 2.52 TOPS on a dense LSTM network. Moreover, it processes a full LSTM for speech recognition with a power dissipation of 41 Watts. Evaluated on the speech recognition benchmark LSTM, ESE is 43x faster and 3x faster than the Core i7 5930k CPU and Pascal TitanX GPU. Compared with the CPU and GPU, the energy efficiency of 40x and 11.5x is improved respectively.

1. **GPU:** GPU stands for Graphics Processing Unit. It is a specialized electronic circuit, which is structured to accelerate the processing of images and graphics. GPUs are hugely parallel and bearing multiple cores that can execute calculations simultaneously. This parallelism allows GPUs well-fitted for deep learning applications, which need a great number of calculations to be performed in simultaneously. GPUs are usually used to accelerate the training and inference of deep neural networks, and they have been acted to produce momentous speedups over conventional CPUs. Now a days, there has been a tendency towards creating specialized hardware architectures for deep learning, for instance ASICs and FPGAs, but GPUs keep a great choice because of their significant performance and comparatively less cost. The paper [2] highlights the effectiveness of deep learning in different types of applications, because of the parallel processing of the GPU platform. The reliability factor hasn't been deeply examined, including the architectural vulnerability of deep learning algorithms and optimizations. This paper describes the evaluation of GPU architecture AVF during deep learning, comparing it with other GPU applications, and explores the optimization techniques. The paper

[2] describes characterizing the architectural vulnerability factor (AVF) of GPU architecture during deep learning, including convolution neural networks (CNNs) and recurrent neural networks (RNNs). It evaluates the AVF of key GPU components, such as the register file, instruction buffer and performance optimization. They modified a GPU simulator to measure AVF, for the specific networks the data execution captured during benchmark tests. The results are about AVF characteristics and assessments of optimization techniques, with a comparison to general-purpose workloads. The paper estimates the failure rate for the register file and instruction buffer based on the AVF results.

1. **ASIC:** ASIC represent Application-Specific Integrated Circuit. It is a one kind of integrated circuit that is structured for a certain application or use case, rather than for common-purpose use. ASICs are hugely reformed and optimized for an individual task, which allows them more workable and faster than general-purpose processors. They are usually utilized in applications, which need high performance, less power consumption, and high levels of integration. ASICs can be structured to include various features, including memory blocks, microprocessors and digital signal processing blocks. But ASICs are generally expensive to design and construct, and they are not as feasible like other kinds of hardware architectures like FPGAs.[3]
2. **SilaGo:** The SiLago Solution is a design methodology for energy-efficient computing and it can be a main use case for specialized hardware for Deep Learning. It allows for reducing the power footprint for modern dark silicon-related problems. In recent years transistors are getting packed more densely onto a chip, and in turn, they are generating more heat. This increase in heat leads to more power consumption and can lead to thermal issues, which can cause the chip to malfunction or even fail. To prevent this, chip manufacturers must limit the amount of power that can be used at any given time, which means that a significant portion of the chip must remain "dark" or unpowered. The concept of Dark Silicon has led to the development of new design methodologies, such as the SilaGo Method, that aim to make better use of the available silicon area and reduce power consumption.[4]

It uses heterogeneous dark silicon aware coarse grain reconfigurable fabrics (CGRA). It consists of a regular array of processing elements (PEs) that are connected by a programmable interconnect network. The PEs are typically

optimized for a specific set of operations, such as arithmetic or logic, and can be reconfigured to perform different operations as needed. The interconnect network allows the PEs to communicate with each other and with external memory and I/O devices. The term "coarse-grain" refers to the fact that the PEs are larger and more complex than those found in fine-grain reconfigurable architectures, such as field- programmable gate arrays (FPGAs) to address the challenges posed by the dark silicon era, where only a small percentage of a chip can be active at any given time due to power constraints. Overall, the SiLago Solution is designed to provide energy- efficient computing solutions that can keep up with the increasing design complexity and low power demands of modern applications requiring deep learning.

## i. The components of the architecture:

SiLaGO is estimated to be the best alternative to tackle the ever-changing problems of deep learning by mitigating the use of interconnects and incorporating "synchorocity" [3] "as a term used to describe the design philosophy behind SiLaGO architecture. It refers to the idea of minimizing the use of interconnects between different components of the hardware design.

In SiLaGO, the sub-components are placed in a way that minimizes the distance to the sub-component connected to it in the next SiLaGO block. This reduces the power consumption in wiring and improves the overall efficiency of the design. By minimizing the use of interconnects, SiLaGO aims to reduce the latency and power consumption of the hardware, which is particularly important for deep learning applications that require high computational power and energy efficiency in designs. SiLaGO blocks are designed to minimize the inter-connecting wires, which reduces power consumption and improves efficiency. The SiLaGO architecture is now going through a surging stage from research to implementation, and some authors have already implemented SiLaGO as a neural network. [5]

## Conclusion

In conclusion, the field of deep learning has seen tremendous growth in recent years, and the demand for high-performance computing architecture has never been greater. ASIC, FPGAs, GPUs, and SilaGO are all viable hardware architectures for deep learning, each with its own unique advantages and disadvantages. ASICs offer high performance and low power consumption, but they are expensive to design and manufacture. FPGAs are flexible and can be reprogrammed for different tasks, but

they are less efficient than ASICs. GPUs are widely used for deep learning due to their high parallelism, but they consume a lot of power. SilaGO is a promising solution that combines the benefits of ASICs and FPGAs, offering high performance, low power consumption, and flexibility. In summary, the choice of hardware architecture for deep learning depends on the specific requirements of the application, and each of these architectures has its own strengths and weaknesses.

## References

1. Wang, C. Wang, X. Zhou and H. Chen, "An Overview of FPGA Based Deep Learning Accelerators: Challenges and Opportunities," in 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), Zhangjiajie,

China, 2019 pp. 1674-1681. doi: 10.1109/HPCC/SmartCity/DSS.2019.00229

1. D. Santoso and H. Jeon, " Understanding of GPU Architectural Vulnerability for Deep Learning Workloads," 2019 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), Noordwijk, Netherlands, 2019, pp. 1-6, doi: 10.1109/DFT.2019.8875404.
2. Bhattacharya, G. (2021). From DNNs to GANs: Review of efficient hardware architectures for deep learning. ArXiv, abs/2107.00092.
3. Hemani, A., Farahini, N., Jafri, S.M.A.H., Sohofi, H., Li, S., Paul, K. (2017). The SiLago Solution: Architecture and Design Methods for a Heterogeneous Dark Silicon Aware Coarse Grain Reconfigurable Fabric. In: Rahmani, A., Liljeberg, P., Hemani, A., Jantsch, A., Tenhunen, H. (eds) The Dark Side of Silicon. Springer, Cham. <https://doi.org/10.1007/978-3-319-31596-6_3>
4. Y. Zhang, “Mapping quantized convolutional layers on the SiLago platform,” Dissertation, 2022. https://www.diva- portal.org/smash/record.jsf?pid=diva2%3A1740699
5. Machupalli, Raju, Masum Hossain, and Mrinal Mandal. "Review of ASIC accelerators for deep neural network." Microprocessors and Microsystems 89 (2022): 104441.
6. Boutros, Andrew, Sadegh Yazdanshenas, and Vaughn Betz. "You cannot improve what you do not measure: FPGA vs. ASIC efficiency gaps for convolutional neural network inference." ACM Transactions on Reconfigurable Technology and Systems (TRETS) 11.3 (2018): 1-23.
7. Manasi, Susmita Dey, and Sachin S. Sapatnekar. "DeepOpt: Optimized scheduling of CNN workloads for ASIC-based systolic deep learning accelerators." Proceedings of the 26th Asia and South Pacific Design Automation Conference. 2021.
8. Hu, Yunxiang, Yuhao Liu, and Zhuovuan Liu. "A survey on convolutional neural network accelerators: GPU, FPGA and ASIC." 2022 14th International Conference on Computer Research and Development (ICCRD). IEEE, 2022.
9. Li, Hengyi, et al. "An architecture-level analysis on deep learning models for low-impact computations." Artificial Intelligence Review 56.3 (2023): 1971-2010.
10. Hao, Yufeng. "A general neural network hardware architecture on FPGA." arXiv preprint arXiv:1711.05860 (2017).
11. Du, Gaoming, et al. "Efficient softmax hardware architecture for deep neural networks." Proceedings of the 2019 on Great Lakes Symposium on VLSI. 2019.
12. Medus, Leandro D., et al. "A novel systolic parallel hardware architecture for the FPGA acceleration of feedforward neural networks." IEEE Access 7 (2019): 76084-76103.
13. Wang, Xiaowei, et al. "Compute-capable block RAMs for efficient deep learning acceleration on FPGAs." 2021 IEEE 29th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM). IEEE, 2021.
14. Zhou, Yuteng, Shrutika Redkar, and Xinming Huang. "Deep learning binary neural network on an FPGA." 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS). IEEE, 2017.
15. Wang, Jichen, Jun Lin, and Zhongfeng Wang. "Efficient hardware architectures for deep convolutional neural network." IEEE Transactions on Circuits and Systems I: Regular Papers 65.6 (2017): 1941-1953.
16. Shawahna, Ahmad, Sadiq M. Sait, and Aiman El-Maleh. "FPGA-based accelerators of deep learning and classification: A review." ieee Access 7 (2018): 7823-7859.
17. Farrukh, Fasih Ud Din, et al. "Optimization for efficient hardware implementation of CNN on FPGA." 2018 IEEE International Conference on Integrated Circuits, Technologies and Applications (ICTA). IEEE, 2018.
18. Wang, Chao, et al. "Service-oriented architecture on FPGA-based MPSoC." IEEE Transactions on Parallel and Distributed Systems 28.10 (2017): 2993-3006.
19. Zaman, Kh Shahriya, et al. "Custom hardware architectures for deep learning on portable devices: a review." IEEE Transactions on Neural Networks and Learning Systems (2021).
20. Wang, Jichen, Jun Lin, and Zhongfeng Wang. "Efficient hardware architectures for deep convolutional neural network." IEEE Transactions on Circuits and Systems I: Regular Papers 65.6 (2017): 1941-1953.
21. Shams, Shayan, et al. "Evaluation of deep learning frameworks over different HPC architectures." 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS). IEEE, 2017.
22. Shi, Shaohuai, Qiang Wang, and Xiaowen Chu. "Performance modeling and evaluation of distributed deep learning frameworks on gpus." 2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber (DASC/PiCom/DataCom/CyberSciTech). IEEE, 2018.
23. Gizopoulos, Dimitris, et al. "Modern hardware margins: CPUs, GPUs, FPGAs recent system-level studies." 2019 IEEE 25th International Symposium on On-Line Testing and Robust System Design (IOLTS). IEEE, 2019.