A Review on AI Chip Design

Rajesh Kumar
M.Tech Scholar
Vidhyapeeth Institute of Science & Technology
Bhopal, M.P., India
srajeshkarn79@gmail.com

Swati Gupta
HOD
Vidhyapeeth Institute of Science & Technology
Bhopal, M.P, India
swatishah03@gmail.com

***Abstract*: In recent years, artificial intelligence (AI) technologies have been widely used in many business areas. With the attention and investment of scientific researchers and research companies around the world, artificial intelligence technologies have proven their irreplaceable value in traditional speech recognition, image recognition, search/recommendation engines, and other areas. At the same time, however, the computational effort for artificial intelligence technologies is increasing dramatically, posing a huge challenge to the computing power of hardware devices. First, in this paper, we describe the direction of AI chip technology development, including the technical shortcomings of existing AI chips. So, we present the directions of AI chip development in recent years.**

Keywords: AI chip, GPUs, CPU, ASICs.

I. INTRODUCTION

An AI accelerator is a class of microprocessors [1] or computer systems [2] that have been developed as hardware accelerators for applications with artificial intelligence, in particular artificial neural networks, image processing, and machine learning. Typical applications are robotics algorithms, the Internet of Things, and other data-intensive or sensor-controlled activities [3]. There are often many basic projects that typically focus on low-precision arithmetic, new data flow architectures, or in-memory computational functions [4]. A typical AI chip for integrated circuits contains billions of MOSFETs [5].

There are a number of manufacturer-specific terms for devices in this category and this is an emerging technology without a dominant design. AI accelerators can be found in many devices such as smartphones, tablets, and computers around the world.

Computer systems have often integrated the processor with special accelerators for special tasks called coprocessors. Application-specific hardware units include graphics cards, sound cards, graphics processors, and digital signal processors.

A. AI Chip Basics

AI chips include graphics processors unit (GPUs), field-programmable gate arrays (FPGAs), and application-specific integrated circuits (ASICs) that specialize in AI. General-purpose chips such as central processing units (CPUs) can also be used for some simpler AI tasks, but processors become less useful as AI progresses. Like generic processors, AI chips gain speed and efficiency (meaning they can perform more computations per unit of power consumed) by containing a large number of smaller and smaller transistors that run faster and use less power than smaller transistors large. Unlike processors, however, AI chips also have other AI-optimized design features. These features greatly accelerate the identical, predictable, and independent computations required by artificial intelligence algorithms. This understands that a large number of calculations are performed in parallel and not one after the other as with CPUs. Calculates low-precision numbers in order to successfully implement AI algorithms but reduces the number of transistors needed for the same calculation; Accelerate memory access by storing, for example, an entire AI algorithm in a single AI chip; and using specially designed programming languages to efficiently translate AI computer code to run on an AI chip [5].

Different types of AI chips are useful for different tasks. GPUs are most commonly used for the initial development and refinement of artificial intelligence algorithms. This process is called "training". FPGAs are primarily used to apply trained AI algorithms to actual data input. This is often called "inference". ASICs can be designed for training or inference.

II. LITERATURE REVIEW

H. Momose et al. [6] in this article, ASIC chips designed for learning functions generalize with competitive computing power. However, the restrictions of Moore's Law have begun to impose themselves on this emerging sort of AI chip, creating the necessity for brand spanking new technological innovations. As for Edge AI chips, research is advancing on data compression technology to scale back power consumption while maintaining high performance.

Y. Chen et al. [7] during this article, we specialise in summarizing recent advances within the design of deep neural network (DNN) accelerators, aka DNN accelerators. We discuss different architectures that support DNN projects with regards to compute units, data flow optimization, targeted network topologies, architectures for brand spanking new technologies and accelerators for brand spanking new applications. We also give our insights into the longer term trend of AI chip designs.

Tso-Bing Juang et al. [8] during this paper, we've proposed an efficient design for zone delay products (ADP) for CNN (Convolutional Neural Network) circuits using logarithmic number systems (LNS). By using LNS-based schemes, the space required for an outsized number of conventional multipliers required in CNN circuits are often greatly reduced. The simulation results show that with ADP savings of nearly 60%, our proposed design can generate fewer errors than the normal multiplier-based design suitable for deep learning applications.

John R Hu et al. [9] this text presented a scientific approach for identifying, predicting and optimizing Design Process Interaction (DPI). And to optimize the general technology to the chip / system. This resulted within the best performance, performance and efficiency for the GPU / SOC for top performance computing (HPC), AI (AI) and autonomous vehicle applications.

III. System-on-Chip (SOC) Architecture

System-on-chip (SoC) architectures increasingly feature hardware accelerators for energy-efficient performance. Complex applications use these special components to enhance the performance of selected processing cores.

For example, hardware accelerators for machine learning applications are increasingly wont to identify the underlying relationships in unstructured big data [10]. Many of those algorithms first create an indoor model by analyzing very large amounts of knowledge . in order that they use this model to form decisions. due to the inherent parallelism of their cores, they're good candidates for hardware specialization, especially in loosely coupled accelerators (LCAs) [11] - [13]. The instance in Figure 1 shows a part of a SoC that has two LCAs and a processor core connected to external memory (DRAM). Each ACL consists of the accelerator logic that implements the compute and personal local storage (PLM), which stores the info that must be accessed with a hard and fast latency. PLMs structure the storage subsystem of the SoC accelerator and are made from many units called PLM elements.

Each of those PLM elements is employed to store an algorithm arrangement. Although PLMs are known to be responsible for most of the accelerator domain [14], they will only contain some of the entire working dataset at any given time, which is fully stored in DRAM. . The accelerator calculation is thus organized in successive iterations during which the info is exchanged step by step through DMA transfers with DRAM.

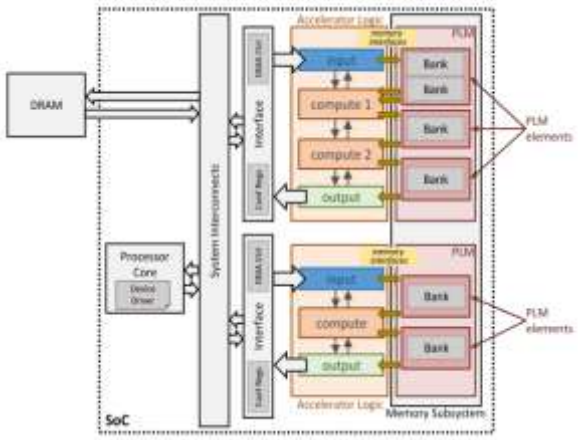

Fig. 1. Accelerator-based SoC

Therefore, the accelerator logic is structured with multiple blocks of hardware operating simultaneously in parallel or within the pipeline (i.e. inputs, computers, and outputs). The inbound and outbound hardware blocks handle data transfers, while hardware blocks implement accelerator functionality. PLM management is therefore completely transparent to the processor core, which is liable for processing the info within the DRAM and controlling the execution of the accelerator. An OS runs on the kernel and every accelerator is managed by a tool driver.

With special microarchitectures for accelerator logic and PLM, lifecycle assessments can perform better than processor cores to run the algorithm that they were developed. Accelerator logic can cash in of spatial parallelism to perform multiple operations in parallel. The dimensions of every PLM item is adjusted consistent with the quantity of knowledge to be archived.

Although processor memories are designed for sequential access (even when freeing memory with the accelerator [15], [16]), PLMs require more ports in order that the accelerator logic can perform more memory operations within the same clock cycle and increase hardware parallelism. There are several solutions to implement multiport memories [17]. Distributed registers, which are fully included within the accelerator logic, are used for little data structures that are accessed frequently.

However, the mixture size of those logs is understood to grow exponentially with the quantity of knowledge to be stored. Large and sophisticated data structures require the allocation of property (IP) blocks of dedicated memory, which are more efficient in terms of resources. However, because the size of those storage elements increases significantly with the amount of ports [18], technology vendors generally only offer storage IP addresses with one or two ports [19].

IV. The Development Direction of AI Chip Technology

A. Technical defects of existing AI chips

At present, the core of mainstream AI chips is to achieve the speedup of the main convolution operation in CNN (convolutional neural network) by multipliers and accumulations. This generation of AI chips mainly has the following three aspects: First, the amount of data required for deep learning calculation is huge, and the memory bandwidth becomes the bottleneck of the entire system. Second, a large amount of memory access and MAC array computing, resulting in the overall power consumption of AI chips increased. Third, deep learning requires a lot of computing power. With the rapid development of deep learning algorithms, new algorithms are not well supported in accelerators that have been solidified. Therefore, the best way to improve computing power is to do hardware acceleration, which is to improve the computing power of AI chips.

B. The breakthrough direction of AI chips in the future

Therefore, it is foreseeable that the next generation of AI chips will have the following five trends.

First, more efficient convolution deconstruction / reuse.

Based on the standard SIMD, CNN can further reduce data communication on the bus due to its special multiplexing mechanism. The concept of reuse is particularly important in very large neural networks. How to reasonably decompose and map these super large convolutions to effective hardware has become a research direction.

Second, lower inference calculation / storage bit width.

One of the biggest evolutions of AI chips may be the rapid reduction of neural network parameters/calculation bit widths—from 32-bit floating point to 16-bit floating point/fixed point, 8-bit fixed point, and even 4-bit fixed point. In the field of theoretical computing, 2 or even 1 bit of parameter width has gradually entered the practice field.

Third, how to reduce the memory access delay.

When computing components are no longer the design bottleneck of neural network accelerators, how to reduce memory access latency will be the next research direction.

Fourth, a more sparse large-scale vector implementation.

Although the neural network is large, there are many cases where zero is input. At this time, the sparse calculation can reduce the useless energy efficiency, so as to reduce the useless power consumption.

Fifth, Computing and storage integration.

Process-in-memory technology, through the new nonvolatile storage device, adds neural network computing function to the storage array, eliminating data moving operation, and realizes the neural network processing of computational storage integration, which significantly improves the power consumption performance.

V. Artificial Intelligence Chip Key Market Segments

A. By Chip Type
- GPU
- ASIC
- FPGA
- CPU
- Others

B. By Application
- Natural Language Processing (NLP)
- Robotic
- Computer Vision
- Network Security
- Others

C. By Technology
- System-on-Chip (SoC)
- System-in-Package (SIP)
- Multi-chip Module
- Others

D. By Processing Type
- Edge
- Cloud

E. By Industry Vertical
- Media & Advertising
- BFSI
- IT & Telecom
- Retail
- Healthcare

- Automotive & Transportation
- Others

VI. CONCLUSION

In recent years, AI technology has made continuous inroads. Being an important physical foundation of AI technology, AI chips have high industrial value and strategic location. However, from a general trend perspective, it is still in the early stages of AI chip development and there is a large margin for innovation in scientific research and industrial applications. Only when the computing power of the core reaches a certain level and the synergy between algorithm and big data can lead artificial intelligence to greater progress.

REFERENCES

[1] M. Horowitz, "Computing's energy problem (and what we can do about it)," in ISSCC Digest of Technical Papers, Feb. 2014, pp. 10–14.

[2] S. Borkar and A. A. Chien, "The future of microprocessors," Communication of the ACM, vol. 54, pp. 67–77, May 2011.

[3] M. Taylor, "Is dark silicon useful? Harnessing the four horsemen of the coming dark silicon apocalypse," in Proc. of the Design Automation Conf., Jun. 2012, pp. 1131–1136.

[4] T. Chen et al., "DianNao: A Small-footprint High-throughput Accelerator for Ubiquitous Machine-learning," in Proc. of Intl. Conf. on Architectural Support for Programming Languages and Operating Systems, 2014, pp. 269–284.

[5] Z. Wang, K. H. Lee, and N. Verma, "Hardware specialization in lowpower sensing applications to address energy and resilience," Journal of Signal Processing Systems, vol. 78, no. 1, pp. 49–62, 2014.

[6] H. Momose, Tatsuya Kaneko and Tetsuya Asai "Systems and circuits for AI chips and their trends" japanese Journal of Applied Physics, Volume 59, Number 5, 2020.

[7] Y. Chen, YuanXie "A Survey of Accelerator Architectures for Deep Neural Networks" Engineering Volume 6, Issue 3, March 2020, Pages 264-274.

[8] Tso-Bing Juang,, Cong-Yi Lin and Guan-Zhong Lin "Area-Delay Product Efficient Design for Convolutional Neural Network Circuits Using Logarithmic Number Systems" ISOCC 2018.

[9] John R Hu, James Chen, Boon-khim Liew, Yanfeng Wang, Lianxi Shen, Lin Cong, "Systematic Co-Optimization From Chip Design, Process Technology To Systems For GPU AI Chip", IEEE, 2018.

[10] G. Grewal, S. Areibi, M. Westrik, Z. Abuowaimer and B. Zhao, "Automatic Flow Selection and Quality-of-Result Estimation for FPGA Placement," 2017 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), Lake Buena Vista, FL, 2017, pp. 115-123.

[11] W. J. Chan, P. Ho, A. B. Kahng, P. Saxena, "Routability Optimization for Industrial Designs at Sub-14nm Process Nodes Using Machine Learning," Proceedings of the 2017 ACM on International Symposium on Physical Design (ISPD 17), pp. 15-21.

[12] B. Yu, D. Z. Pan, T. Matsunawa and X. Zeng, "Machine learning and pattern matching in physical design," The 20th Asia and South Pacific Design Automation Conference, Chiba, 2015, pp. 286-293.

[13] G. Batra, Z. Jacobsen, N. Santhanam, "Improving the semiconductor industry through advanced analytics", McKinsey Article, March 2016.

[14] D. K. May, "Improving Verification Predictability and Efficiency Using Big Data,", DVCon Proceedings, February 2018.

[15] C. Zhang et al., "Optimizing FPGA-based Accelerator Design for Deep Convolutional Neural Networks," in Proc. of the Int. Symp. on FieldProgrammable Gate Arrays, 2015, pp. 161–170.

[16] E. Cota et al., "An analysis of accelerator coupling in heterogeneous architectures," in Proc. of the Design Automation Conf., Jun. 2015, pp. 1–6.

[17] J. Cong et al., "Architecture support for accelerator-rich CMPs," in Proc. of the Design Automation Conf., 2012, pp. 843–849.

[18] B. Li, Z. Fang, and R. Iyer, "Template-based memory access engine for accelerators in SoCs," in Proc. of the Asian and South-Pacific Design Automation Conf., Jan 2011, pp. 147–153.

[19] M. Lyons et al., "The accelerator store: A shared memory framework for accelerator-based systems," ACM Trans. on Architecture and Code Optimization, vol. 8, no. 4, pp. 48:1–48:22, Jan. 2012.