SRDA: Next-Generation AI Computing Architecture Whitepaper

MoonQuest-AI

research@moonquest.cn

Abstract

With the rapid development of Artificial Intelligence (AI) technology, especially the rise of Large Language Models (LLM) and generative AI, the demand for computing power is showing unprecedented growth. Traditional computing architectures are gradually exposing their inherent limitations when faced with the sharp expansion of AI model scale and complexity. We believe that many seemingly different computing problems (memory wall, storage wall, power consumption wall, etc.) are fundamentally related to I/O problems (data read/write and movement), which restricts the full utilization of theoretical computing power.

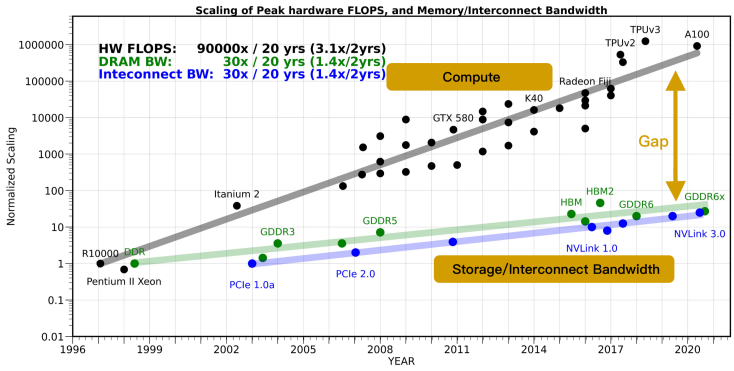


Figure1. Scaling of Peak hardware FLOPS, and Memory/Interconnect Bandwidth

This whitepaper aims to introduce an innovative computing architecture designed specifically for AI computing – SRDA, which can stand for System-level Reconfigurable Dataflow Architecture (not only chip-level dataflow, but data center-level reconfigurable dataflow, with unique innovations in cluster interconnection), or Simplified Reconfigurable Dataflow Architecture (extremely simplified). Originating from profound I/O technology accumulation, SRDA, with innovations in several key technologies such as a distributed 3D memory system, converged network technology, and reconfigurable dataflow, achieves a minimalist and efficient hardware-software converged architecture, dedicated to systematically solving the core challenges currently faced by AI infrastructure. This paper will elaborate on SRDA's architectural philosophy, technical advantages, and its initial core components, discuss how it overcomes existing compute bottlenecks from an I/O optimization perspective, and look forward to its potential impact on the future development of AI computing system.

Keywords

3D-DRAM, System-level Dataflow, Converged Network, Compute-Storage-Interconnect Convergence, Simplified Software Stack, Reconfigurable

1 Introduction

Artificial intelligence, especially deep learning, has made significant progress in the past decade and has shown enormous application potential in many fields. The emergence of AI models, particularly architectures like Transformer, has caused model parameters to surge from millions to trillions. The exponential growth in model scale has directly led to extreme demands for computing resources, especially parallel processing capabilities, memory bandwidth, and communication performance.

However, Moore's Law and Dennard scaling, which have been relied upon for many years, are approaching their limits or have become ineffective. In the pursuit of higher computing power, traditional computing architectures have encountered multiple constraints such as the "power wall," "memory wall," and "cost wall." Specifically:

- **Von Neumann Bottleneck**: Frequent data movement caused by the separation of computing units and storage units severely limits actual performance and energy efficiency.
- **Memory Bandwidth Limitation**: Although existing mainstream GPGPU architectures continuously improve theoretical peak computing power, their multi-level shared Cache memory architecture is prone to data read/write congestion during concurrent multi-threaded access. This often makes memory bandwidth a bottleneck in practical applications, especially when processing giant AI models.
- **Insufficient Computing Power Utilization**: Due to inherent architectural limitations, a large number of communication tasks occupying core computing resources, and memory access bottlenecks, among other factors, the theoretical peak computing power of existing chips cannot be fully utilized in actual AI workloads.
- **Software Ecosystem and Ease of Use**: The software stacks of existing AI acceleration solutions are mostly complex, with high migration and optimization costs, limiting their widespread application.
- **Power Consumption Bottleneck**: The power consumption metrics of top-tier AI accelerator chips are constantly setting new highs, becoming a core bottleneck restricting their wider application and sustainable development. Taking NVIDIA H100 GPU as an example, its typical board power consumption is as high as 700 watts. The super-node cluster solution based on optical modules is even more power-hungry. This is because GPGPU-like architectures mainly rely on directly stacking computing cores to improve computing power. Currently, electricity costs often account for 30% to 60% of the total operating expenses (OpEx) of data centers, and chip power consumption is one of the primary concerns for customers.
- **Large-Scale Cluster Expansion Difficulties**: With the increasing demand for computing, building and managing large-scale AI clusters composed of thousands of nodes faces problems such as complex network topology, high communication latency, poor reliability, and high costs. In particular, the design of traditional two-layer networks (scale-up network like NVLink, and scale-out networks like InfiniBand/Ethernet) brings about issues such as bandwidth tier

differences, increased latency due to protocol conversion overhead, and complex communication management.

Faced with these challenges, the industry urgently needs innovative computing architectures to break through existing bottlenecks and meet the needs of continuous AI technology development.

2 Key Features In Next-Generation AI Computing Architecture

To effectively support the continuous evolution of AI technology and overcome existing bottlenecks, we believe the next-generation AI computing architecture should possess a series of key features. These features must not only address current performance and efficiency issues but also provide a flexible and scalable foundation for future AI models and applications.

- **Native Dataflow Processing Capability**: The essence of AI computing is the flow and transformation of data in complex computational graphs. The next-generation architecture should be able to natively and efficiently express and execute dataflows, minimizing unnecessary data movement, instruction decoding, and control overhead, thereby achieving tight coupling between computation and dataflow.
- **Innovative Memory System Design and Ultra-High Bandwidth**: With model parameters exceeding trillions, the architecture must provide memory capacity and bandwidth far beyond current levels. The next-generation architecture should adopt advanced memory technologies such as 3D stacked DRAM and innovate memory organization methods to build a distributed, high-bandwidth on-chip memory network privatized for computing units. This aims to fundamentally solve problems like severe data congestion, high access conflicts, and low bandwidth utilization caused by traditional architectures (such as the multi-level shared Cache memory architecture of GPGPU), ensuring ample and efficient data supply.
- **Integrated Converged Network**: The traditional two-layer network architecture used in data centers, with separate intra-node high-speed interconnects (Scale-up, e.g., NVLink) and inter-node networks (Scale-out, e.g., InfiniBand/RoCE), faces challenges such as complex management, uneven bandwidth utilization, and a large amount of computing resources being used for communication auxiliary tasks like cross-network domain data forwarding, aggregation, and protocol conversion. The next-generation architecture should aim to build a unified converged network, breaking the boundaries of the two-layer network. By using integrated I/O Dies or dedicated communication coprocessors, communication management tasks can be offloaded from the main computing units, providing a unified, efficient, and low-latency global communication view for the computing units. This includes hardware-level support for flexible packet forwarding, multicast, reduction operations, and simplifying software programming complexity through memory semantic interfaces, thereby truly achieving decoupling and efficient collaboration between computation and communication.
- **Extreme Training and Inference Efficiency**: Using chain-of-thought reasoning data as training data has become a consensus, and there will be more collaborative needs between the inference phase and the training phase. Therefore, the future chip architecture needs to optimize both training and inference stages simultaneously. For training, the key is to achieve high throughput, high parallel efficiency, and low cost; for inference, the focus is on low latency, high concurrency, and energy efficiency.
- **Advanced Low-Precision Computing Support**: To improve computing density and memory efficiency, supporting and optimizing low-precision data formats (such as FP8, FP4, etc.) for computation is crucial. This includes hardware-level native support for low-precision

operations, providing fine-grained quantization (tile-wise/block-wise quantization) and high-precision accumulation.

- **Deep Collaboration and Optimization of Computation and Communication**: In large-scale distributed computing, communication overhead is one of the main bottlenecks. The architecture should achieve a high degree of overlap between computation and communication through hardware-software co-design to hide communication latency. This involves efficient on-chip/inter-chip/inter-node interconnection technologies.
- **Highly Flexible Model Mapping and Reconfigurability**: AI algorithms and model structures are iterating rapidly. The next-generation architecture should possess high flexibility and reconfigurability to adapt to new models, operators, and even entirely new computing paradigms, avoiding premature obsolescence due to hardware solidification. This requires the architecture to be more than just a fixed-function ASIC, but to have a certain degree of software-configurable reconfigurable computing capability.
- **Full-Stack Optimization and Ease of Use**: Achieving full-stack converged optimization from algorithms, compilers, and drivers to hardware is key to unlocking the potential of the architecture. A powerful compiler can automatically and efficiently map models described in high-level languages to the underlying hardware, shielding hardware complexity and reducing users' adoption and migration costs.
- **Scalability and Cost-Effectiveness**: The architecture should support smooth, linear scaling from a single node to ultra-large-scale clusters, while achieving the best balance between performance, power consumption, and total cost of ownership (TCO), making advanced AI computing power more economically viable.

The design of the SRDA architecture revolves around these key features, aiming to provide a solid computing power foundation for the future development of AI.

3 SRDA Architecture: A New Paradigm for AI Acceleration Centered on Dataflow

SRDA (Simplified Reconfigurable Dataflow Architecture / System-level Reconfigurable Dataflow Architecture) is an AI computing architecture centered on dataflow and designed with hardware-software hyper-convergence. Its core design philosophy is to maximize the efficiency, flexibility, and scalability of AI computing through simplification and reconfigurability.

3.1 Design Philosophy: Data-Centric, Collaborative Optimization

The design philosophy of SRDA is rooted in a deep understanding of the characteristics of AI computing workloads and is embodied through the following four core principles, aiming to fundamentally address the bottlenecks of existing computing architectures in AI applications:

System-level Dataflow Architecture

AI computation, especially the training and inference of deep neural networks, is essentially a process of large-scale, parallelized data flowing and transforming between computation nodes according to a specific computational graph. Under traditional “Control-Flow” Architecture, the sequential execution of instructions and complex memory hierarchy access often become performance bottlenecks, leading to idle computing units and unnecessary data movement.

The SRDA architecture, however, takes "Data-flow" as its first principle.

In chip level, it directly maps the data dependency relationships in statistic AI computational graphs through hardware design. Intermediate data is transmitted directly point-to-point between computing units over optimized, customizable computation paths, significantly reducing reliance on memory and access frequency to memory.

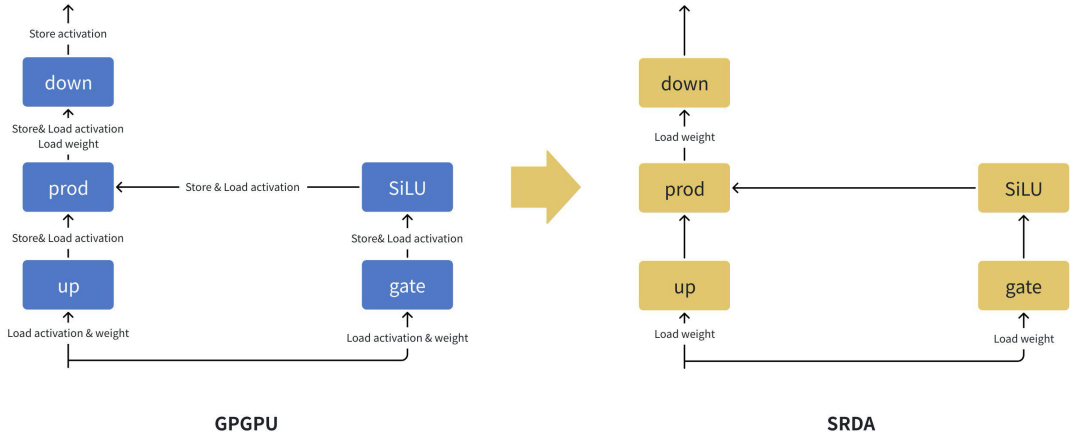


Figure 2. GPGPU vs SRDA on MLP

The figure illustrates the computational process of the Multi-Layer Perceptron (MLP), a module that accounts for a significant portion (40%-60%) of the execution time during both the training and inference phases of large-scale models. In conventional architectures, computations are performed on a kernel-by-kernel basis. Each kernel execution necessitates reading the required data from memory and subsequently writing the computed results back to memory. This process incurs substantial, often redundant, memory read/write operations. In contrast, SRDA obviates the need to write intermediate computational results to memory. Instead, these results are directly streamed to the subsequent computational unit. This approach effectively mitigates the repeated read/write cycles for intermediate data, thereby significantly alleviating memory bandwidth pressure and concurrently enhancing computational efficiency.

Furthermore, SRDA extends this dataflow philosophy to the system level (cluster, data center). Through converged network technology and its synergistic compiler, SRDA aims to achieve efficient, low-latency, on-demand data flow across multiple chips and server nodes. This means that not only are on-chip data paths reconfigurable, but the data pathways and resource scheduling within the entire cluster also embody the characteristics of "software-defined dataflow". This enables global optimization based on the computation and communication patterns of large-scale distributed training or inferencing tasks (such as tensor parallelism and expert parallelism, and pipeline parallelism).

This end-to-end dataflow optimization, from chip to system, fundamentally reduces the distance and frequency of data movement – one of the primary performance and energy consumption bottlenecks in current computing systems. SRDA's fine-grained management of I/O paths ensures efficient data supply to compute units. By making data "flow" and letting computation "follow" the data, SRDA aims to maximize the proportion of effective computation and minimize waiting and movement overhead, thereby enhancing performance while reducing power consumption caused by data movement.

Simplicity and Efficiency

Designed specifically for the AI computing workloads, SRDA chooses a "minimalist" path, stripping away complex control logic, redundant instruction sets, and multi-level cache coherence mechanisms introduced in traditional general-purpose processors to support diverse tasks. As an AI

Domain Specific Architecture (AI-DSA) for large AI models, SRDA focuses hardware resources more on the core operations of AI computation itself, such as tensor operations and vector processing.

This architectural simplicity brings multiple benefits:

- **Higher chip area efficiency:** More effective computing units and on-chip memory can be integrated within the same chip area, directly increasing raw computing power.
- **Higher energy efficiency:** Reduces energy consumption from non-computation-related transistor activity and complex control logic, allowing each watt of electricity to be more effectively converted into useful computing power.
- **Lower complexity:** A simplified architecture also means more manageable design, verification, and manufacturing complexity. For example, the underlying system is based on the open-source RISC-V instruction, further simplifying the instruction system and operator development difficulty. SRDA pursues a "just right" hardware design, i.e., meeting the core needs of AI computation in the most direct and efficient way.

Reconfigurability and Adaptability

SRDA is a specialized architecture designed for the AI field, but it is not rigid fixed-function hardware. AI algorithms and models themselves are rapidly evolving, from classic CNNs and RNNs to mainstream Transformers, Diffusion, and then to emerging MoE (Mixture of Experts), DiT (Diffusion Transformer), ViT (Vision Transformer), Mamba (State Space Models) etc., each with different computational characteristics and dataflow patterns. One of SRDA's key features is its "software-defined" reconfigurability. This means users can adjust data flow paths during computation, the functional combinations of some computing units, and memory access patterns according to their own model architecture. This reconfigurability allows SRDA to:

- Provide customized optimal deployment solutions: SRDA's reconfigurable capability allows for customized deployment under different model architectures, scales, workload patterns, and hardware configurations, maximizing computational efficiency.
- Adapt to future model architecture changes: SRDA's reconfigurability improves hardware-level generalization. In the current era of rapid model development, SRDA allows users to support new model architectures by redefining hardware through software. To simplify this process, users only need to extract and generate the corresponding model's computation graph using SRDA's computation graph generation tool, and the SRDA compiler can automatically complete operator optimization and mapping based on this graph.

Hardware-Software Hyper-Converged Design

To fully unleash hardware potential, SRDA emphasizes the hyper-converged design of hardware architecture and software systems (compilers, drivers, runtime libraries) from its inception. This is not simply developing software for existing hardware, but rather taking software simplicity and ease of use as core indicators during the architecture definition phase.

SRDA's compiler has precise awareness of the hardware's reconfigurable features, the layout and access characteristics of the QDDM™ distributed memory system, and the topology and bandwidth characteristics of the QLink™ interconnect network. This enables the compiler to perform global, static optimization during the compilation phase, efficiently mapping the computation graph to physical computing units, precisely planning data transmission paths on-chip, inter-chip, and inter-node, minimizing memory read/write, and automatically achieving computation and

communication overlap. This deep hyper-converged design allows SRDA to achieve an optimization level that is difficult for traditional GPGPU architecture to reach (with general-purpose compilers) , thus more fully translating the theoretical performance of the hardware into actual performance.

At the same time, the software stack is also committed to compatibility with mainstream AI frameworks (such as PyTorch, JAX, etc), enabling developers to efficiently utilize SRDA's powerful capabilities through familiar methods.

3.2 Core Technology and Innovations

SRDA represents a system-level innovation in computing architecture, its implementation relying on a series of tightly integrated hardware and software innovations:

3.2.1 Chip Level: Significantly Improving Computing Power Utilization

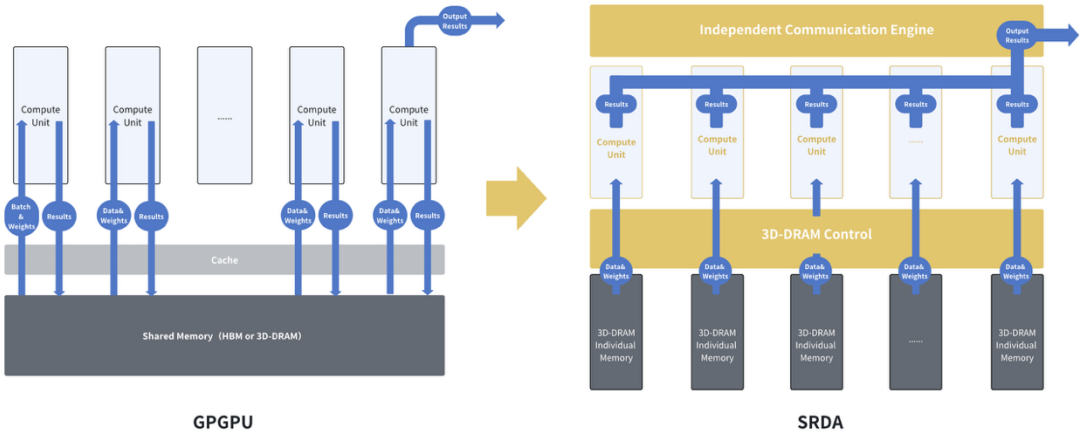


Figure 3. GPGPU vs SRDA on Memory Architecture

A. Simplified RPU (Reconfigurable Processing Units)

At the core of SRDA is its highly parallelized array of compute units. These compute units are not fixed ALUs but are designed as reconfigurable functional units capable of efficiently executing common tensor operations, vector operations, and other specific operations in AI.

- **Reconfigurable Data Paths**: The connections between compute units, as well as the data paths within units, can be configured by the compiler according to the computation graph of a specific AI model, forming dataflow paths optimized for specific tasks.
- **Multi-Precision Support**: Natively supports multiple data types such as FP32, FP16, FP8, FP6, FP4, allowing developers to trade off between performance and precision based on model requirements. For low-precision data types, the hardware natively provides efficient fine-grained online quantization support and high-precision accumulation capabilities.

B. Distributed 3D Stacked Memory Management Technology

To break through the "memory wall" and fundamentally solve the congestion problem of traditional shared memory, SRDA adopts distributed memory technology. This is not just an application of memory technology, but a revolution in memory system architecture.

- **3D-DRAM Integration for Ultra-High Memory Bandwidth**: SRDA fully leverages the technological advantages of advanced 3D stacked memory like 3D-DRAM, directly

integrating an ultra-high bandwidth, large-capacity distributed memory network on the computing chip through 3D stacking processes.

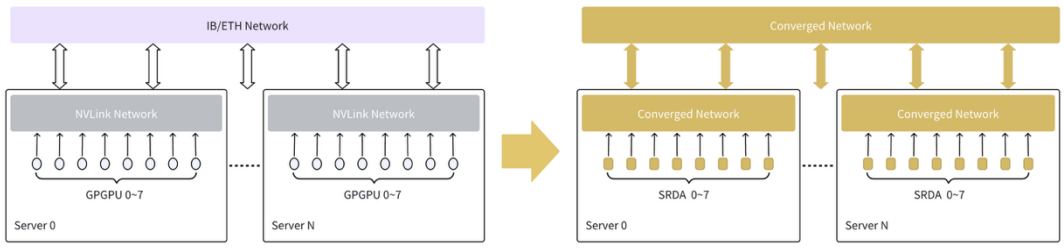
- **Private Dedicated Memory for Compute Unit to Eliminate Congestion**: Unlike traditional GPGPUs that rely on shared memory, which leads to data read/write congestion when multiple compute cores access concurrently, SRDA's core innovation lies in its distributed memory architecture design. Each compute core has its own dedicated, tightly coupled memory. This design allows data access to be completed locally, completely eliminating data congestion caused by competition for shared buses and memory, ensuring that each compute unit can obtain stable, high-bandwidth memory access capabilities.
- **Simplified 3D-DRAM Management Technology**: To fully exploit the advantages of distributed 3D-DRAM, SRDA integrates a simplified 3D-DRAM management technology, which can effectively simplify data transmission paths, shorten access latency, further improve bandwidth utilization, and is specially designed with various data processing and acceleration functions.

SRDA not only achieves extremely high memory bandwidth and capacity but, more importantly, solves the memory access bottleneck plaguing traditional architectures through innovative distributed and privatized design, providing a solid hardware foundation for the efficient operation of the upper-level dataflow.

C. Independent Communication Engine

The QLink™ Converged Interconnect Network integrates a self-developed independent communication engine and enhanced network modules. This aligns with future hardware trends identified by research such as DeepSeek, which involves offloading communication management (such as data forwarding, aggregation, reduction operations, memory layout management, data type conversion, etc.) from the main computing units (like GPU SMs). By using dedicated hardware to handle communication tasks, SRDA achieves computation and communication decoupling, allowing AI core computing units to focus on their primary tasks, avoiding the occupation of valuable computing power by communication affairs, thereby significantly improving overall computing power utilization.

3.2.2 Cluster Level: Converged High-Speed Interconnect Network



Two-Layer Network
NVLink + IB

One-Layer
Converged High Speed Network

Targeting the complexity and efficiency bottlenecks brought by traditional two-layer networks (e.g., NVLink + InfiniBand), SRDA introduces converged high-speed interconnect technology QLink™, aiming to build a highly converged network system from chip-internal data to inter-chip data, inter-node data, and inter-rack data.

- **Converged Network, Simplified System**: The core idea is to merge the traditionally separate intra-node network (Scale-up) and inter-node network (Scale-out) into a unified single-layer network. This design significantly simplifies the network topology of large-scale clusters, reduces protocol conversion and management overhead between multiple network layers, and can effectively decrease the number of backend network ports, thereby lowering deployment costs and complexity.
- **Independent Communication Engine, Freeing Up Compute Unit Resources**: SRDA integrates an independent communication engine and enhanced network modules, offloading communication management (such as data forwarding, aggregation, reduction operations, memory layout management, data type conversion, etc.) from the main computing units (such as GPGPU SMs). By using dedicated hardware to handle communication tasks, SRDA achieves complete decoupling of computation and communication, allowing AI core computing units to focus on their primary tasks, avoiding the occupation of valuable computing power by communication tasks, and thus greatly improving overall computing power utilization.
- **High Bandwidth, Low Latency, and Smart Scheduling**: QLink™ can provide an optimal interconnection bandwidth. More importantly, it integrated self-developed network modules aim to optimize end-to-end communication latency across multiple network layers. also plans to include support for dynamic traffic priority policies and more intelligent routing algorithms to address challenges in complex communication patterns.
- **Simplified Programming Interface and Hardware Synchronization**: SRDA is committed to reducing the complexity of large-scale parallel programming. QLink™, in conjunction with ISA instruction set level optimizations, supports memory semantic communication, avoiding complex software synchronization mechanisms (such as RDMA completion event management) in traditional network programming, thereby reducing software development difficulty and communication latency. This aligns with the industry's expectation of simplifying software stacks through hardware synchronization primitives.

Through converged network design, SRDA not only enhances communication performance but also addresses the complexity and inefficiency issues of traditional multi-layer networks at the system level, laying the foundation for building truly efficient and collaborative large-scale AI computing clusters (100+ racks).

3.2.3 System Level: Quest™ Software Stack, Extremely Simplified, Hardware-Software Co-design

Under traditional computing architectures, the system relies on operator-level optimization to improve performance, thus requiring huge investments in operator development and optimization. Furthermore, migrating operators between different device models is difficult. For instance, when new devices are launched, significant time and human resources are needed for operator optimization to unleash the expected performance of the new hardware.

Compared to traditional architectures, SRDA's storage subsystem has determinism, due to SRDA's compute unit memory privatization and cacheless design. Memory access determinism allows operator developers to fully understand the hardware state at every moment, without needing to consider resource competition and cache reuse in various complex scenarios, greatly reducing the cost of operator development.

Based on the above advantages, the Quest™ software stack consists of the following parts:

- **Access Layer**: The Quest™ access layer supports mainstream AI/ML frameworks, including PyTorch, TensorFlow, JAX, etc., allowing users to continue using familiar programming interfaces to use or migrate to SRDA almost seamlessly. The access layer extracts the computation graph from the AI/ML framework and transforms it into an SRDA IR recognizable by the compiler.
- **Dataflow Compiler**: The Quest™ dataflow compiler accepts the SRDA IR passed from the access layer, combines it with highly optimized operator libraries and communication libraries for optimization, parallel settings, etc., and further generates a dataflow graph mapped to SRDA's physical compute units, memory units, and interconnection network.
- **Operator Development Toolchain**: The Quest™ operator compiler is extended based on the RISC-V toolchain, supports custom operator requirements, and opens up underlying hardware control capabilities to users. At the same time, Quest™ also provides visualization tools to help users identify current performance bottlenecks.

The main advantages of the Quest™ software stack are:

- Low Cost: For most users, the Quest™ access layer supports mainstream AI/ML frameworks like PyTorch, requiring no additional learning cost. For expert users, SRDA's architectural features and the toolchain provided by Quest™ greatly reduce the difficulty of custom operator development and deep optimization.
- High Performance: The dataflow compiler can optimize the SRDA IR from the access layer and transform it into a deployable dataflow graph. For mainstream models, Quest™ also provides highly optimized pre-compiled results, offering users out-of-the-box high-performance training/inference capabilities.

3.3 SRDA Advantages

Overall, through the innovations mentioned above, the SRDA architecture is committed to providing the following core advantages for AI large model training and inference scenarios:

- **Extreme Performance**: Achieves ultra-high memory bandwidth and low-latency access by using 3D stacked memory and compute unit memory privatization, reducing congestion issues of traditional shared memory. It also enhances computing power utilization through reconfigurable dataflow, an independent communication engine, and a converged design of storage, computation, and network, significantly reducing data movement and communication waiting times.
- **Extreme Cost**: Aims to provide superior overall TCO by increasing single-chip/single-node computing power utilization, ultra-low power consumption, simplified networking, a simplified software stack, reduced cluster construction and maintenance complexity, and the adoption of mature domestic manufacturing processes.
- **High Stability**: QLink™ technology, through its converged network design, simplifies network deployment, reduces interconnection costs, and supports the construction of highly stable, cost-effective ultra-large-scale AI computing clusters or super-node solutions.
- **Flexible Model and Algorithm Adaptability**: Reconfigurable dataflow and support for multiple data precisions and fine-grained online quantization enable SRDA to flexibly adapt to continuously evolving AI models and algorithms.**Simplified and Easy-to-Use Software**
- **Development and Migration**: A software stack compatible with mainstream frameworks that

simplifies underlying development, lowering the entry barrier for users.

4 Next Step: More Possibilities for Future AI Hardware

The current design of the SRDA architecture lays a solid foundation, and its future evolution will be closely linked to the development trends of next-generation and even succeeding AI hardware technologies, particularly focusing on deep integration and innovation around higher bandwidth, lower latency, and more intelligent interconnection technologies.

4.1 Optoelectronic Converged Interconnect: Breaking Through Electronic Interconnect Bottlenecks

Current AI cluster scales are increasingly vast, placing extreme demands on inter-node communication bandwidth and latency. Traditional electrical interconnect technologies are gradually facing challenges such as power consumption and signal integrity in long-distance, high-bandwidth scenarios. Silicon photonics technology, especially Optical Circuit Switching (OCS), offers a revolutionary approach for building next-generation data center-level high-bandwidth domains (HBDs).

- **OCS-based Optoelectronic Transceiver Modules (OCSTrx)**: Future products of the SRDA architecture are expected to integrate or be compatible with OCS optoelectronic transceiver modules based on silicon photonics technology. These modules, by embedding low-cost optical switching capabilities into each optoelectronic transceiver, can achieve dynamic, reconfigurable point-to-multipoint optical path connections.
- **Combination of SRDA and OCSTrx**: SRDA's QLink™ independent communication engine can work synergistically with the control plane of OCSTrx. QLink™ will be responsible for high-level communication protocols and task scheduling, while OCSTrx will quickly establish and tear down optical paths according to QLink™'s instructions. This combination will enable unprecedented efficiency for inter-node communication in SRDA clusters, especially for communication-intensive tasks such as Tensor Parallelism (TP) and Expert Parallelism (EP).

4.2 Integration of Future-Oriented Hardware Capabilities

In addition to innovations at the network layer, the SRDA architecture will also continue to absorb and integrate other hardware technologies conducive to improving AI computing efficiency:

- **Deeper Integration of Compute, Memory, and Network**: Further promote the fusion of computing units and memory, exploring closer-proximity, lower-power in-memory computing paradigms to further eliminate data movement and power consumption bottlenecks.
- **Fine-grained Heterogeneous Computing**: On the basis of SRDA's reconfigurable compute units, and according to AI algorithm development trends, more diverse and fine-grained specialized computing logic may be integrated to cope with potentially new operators in future models.
- **Inherent Security and Trusted Computing**: As AI applications delve into critical domains, the requirements for the security and trustworthiness of the computing process will increasingly grow. Future SRDA chips will integrate hardware-level security features.

Through synergistic evolution with these cutting-edge hardware technologies, the SRDA architecture aims to build a continuously leading, highly intelligent, and extremely efficient AI computing power platform, providing an indestructible computing power cornerstone for achieving the grand goal of Artificial General Intelligence (AGI).

The SRDA System-level Reconfigurable Dataflow Architecture, through its I/O-centric breakthrough dataflow-driven design philosophy, hardware-software hyper-convergence, and key technological innovations in computing, memory (especially the innovative distributed 3D stacked memory system), and converged interconnect networks (represented by QLink™), provides a systematic solution to address current and future AI computing power challenges. We not only focus on single-point performance improvement but also on the overall system efficiency, scalability, and cost-effectiveness, emphasizing the potential for performance breakthroughs through architectural innovation under mature processes. We hope SRDA will play a role in promoting the popularization of AI technology, empowering next-generation AI applications, exploring the development of computing architectures, and building an autonomous and controllable AI computing power infrastructure, thereby contributing to the accelerated arrival of the intelligent era.

Acknowledgements

Our sincere gratitude extends to Distinguished Professor Zhongfeng Wang of Nanjing University and his research team; the School of Integrated Circuits at Guangdong University of Technology; and Research Professor Guohui Ding of the Institute for Digital Health at the International Human Phenome Institute (Shanghai), and the Intelligent Medicine Institute at Fudan University and his research team for their support to this project.