Addressing DRAM Performance Analysis Challenges for Network-on-Chip (NoC) Design

Frank Schirrmeister
Arteris
Campbell, CA, USA
frank.schirrmeister@arteris.com

Rocco Jonack
Arteris
Campbell, CA, USA
rocco.jonack@arteris.com

Michael Frank
Arteris
Campbell, CA, USA
frank.schirrmeister@arteris.com

ABSTRACT

Modern System-on-Chip (SoC) architectures and chiplet-based designs often re-use and connect 100's computing and interface blocks, some of which will share caches and external memories in a cache-coherent fashion. Network-on-chip (NoC) implementations can take up 10 to 12 % of chip area, significantly impacting performance, power consumption, and cost [1]. DRAM performance plays a significant role in system performance, and its impact needs to be understood as early as possible during a design flow.

This presentation will discuss different protocol options for RISC-V-based SoCs in the context of their unique advantages in terms of flexibility, performance, and coherency. We introduce NoC development frameworks for cache-coherent applications like AMBA CHI and ACE [2] and non-coherent applications, enabling better architecture optimization and management of physical constraints and reducing interconnect area and power consumption. We will compare critical aspects of the different NoC protocols AMBA 3 AXI, AMBA 4 AXI and ACE, AMBA 5 CHI, and TileLink and discuss the impact of chiplet-based design on NoC protocols. We will also discuss the challenges caused by floor planning and physical layout on NoC topology development with approaches that can achieve up to 5X shorter turn-around time than manual iterations.

Using concrete examples, we will discuss the effect of DRAMs on the system and NoC performance, and we will discuss integration challenges and solutions between development frameworks like Arteris FlexNoc™ [3] and NCore™ [4] with Synopsys Platform Architect [5] and Fraunhofer DRAMSys [6].

1 Comparison of NoC Protocols

NoCs are communication architectures that connect multiple processing elements or IP blocks within a system-on-chip (SoC) or multi-core system. They provide high-bandwidth and low-latency communication channels for efficient data transfer between different components. Designers can customize NoC-based protocols for multi-core SoCs to offer flexible topologies, such as mesh, torus, or tree, and advanced features like adaptive routing, Quality of Service (QoS), and fault tolerance, dividing the network into routers, channels, and interfaces, enabling scalable and modular designs.

Some existing standards for NoC protocols include AMBA 3 AXI (Advanced eXtensible Interface) for high-performance and low-latency communication, offering separate read and write channels, out-of-order transaction completion, and burst transfer support, AMBA 4 introducing ACE (AXI Coherency Extensions), enabling system-level cache coherency in multi-core designs with features such as barrier transactions and cache maintenance operations, AMBA 5 with the CHI (Coherent Hub Interface) protocol, offering traits like virtual networks, Quality of Service (QoS), and fault tolerance for high scalability and performance.

In addition, with a focus on RISC-V-based SoCs, TileLink offers a high-performance, cache-coherent NoC protocol, providing a modular and scalable approach, supporting various coherency models and interconnect topologies. TileLink reduces overhead by supporting atomic operations and implementing configurable address mapping and cache policies.

2 DRAM Impact on NoC Performance

DRAM performance is critical for NoC development because efficient memory access is crucial for data sharing and synchronization between different components within the system. DRAM read and write performance ensures smooth and efficient communication across the network.

NoCs employ buffers at various network nodes to temporarily store and manage incoming and outgoing data packets, often implemented using DRAM-based memory structures. DRAM performance impacts buffer read and write operations, reducing congestion and ensuring smoother flow control. NoCs also often support virtual communication channels, each with specific quality-of-service (QoS) requirements. DRAM performance impacts the ability to meet these QoS requirements. Faster DRAM access enables more efficient

allocation of bandwidth to different communication channels, ensuring timely delivery of packets and meeting latency or throughput targets. As the number of processing elements or IP blocks increases, the demand for memory bandwidth also grows. Higher DRAM performance allows the NoC to handle increased communication traffic and support larger-scale systems without becoming a bottleneck. It ensures that the NoC can effectively scale to accommodate the requirements of complex and interconnected designs. Finally, DRAM performance also has implications for power consumption in NoC designs. Efficient memory access reduces the energy spent on data transfer and communication. Lower latency and higher bandwidth help reduce idle time and allow for more efficient utilization of the network resources, ultimately contributing to improved energy efficiency in the overall system.

3 Flow Integration Examples

The Arteris development frameworks for NoC IP are integrated with performance analysis and digital implementation as shown in Figure 1.

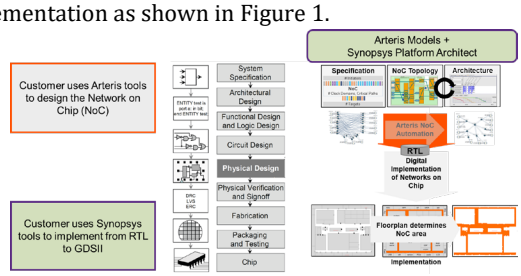

Figure 1: NoC Development Flow, EDA flow diagram sourced from [7]

After importing NoC IP into Synopsys Platform Architect, developers can perform extensive traffic analysis using a Bus Library framework as indicated in Figure 2, focusing on DRAM memory controllers that impact memory access mapping, command generation and timing control, memory organization, configuration, and error correction.

In contrast, memory models manage data storage, access, retrieval, refresh, and retention. For that purpose, the Arteris development frameworks for NoC IP connect to Fraunhofer DRAMSys, a flexible, fast, and open-source DRAM subsystem design space exploration framework based on SystemC TLM-2.0. DRAMSys uses models that reflect the DRAM functionality, power, and temperature, see Figure 3.

The connection allows system designers to analyze the limiting parameters and issues concerning current DRAM standards in the context of system and NoC architectures.

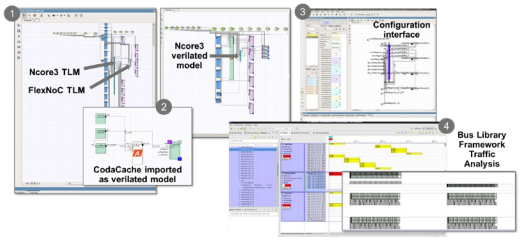

Figure 2: NoC/DRAM Traffic Analysis connecting Arteris and Synopsys development environments

Arteris and Fraunhofer IESE have shown NCore and FlexNoC architecture analysis interoperability with DRAMSys, allowing for more realistic performance analysis before committing to NoC architectures, optimizing performance, power, and cost.

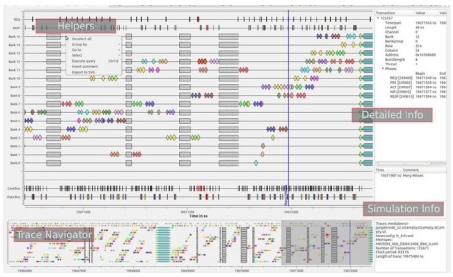

Figure 3: The DRAMSys Trace Analyzer helps to examine the behavior of an application with respect to its DRAM accesses. This includes, e.g., bank level parallelism and command and data bus load.

3 Summary

DRAM performance is critical for today's advanced SoC and chiplet based architectures due to requirements on efficiency of data movement, computation, AI model complexity, real-time inference, and energy consumption. Early integration of NoC development with DRAM performance analysis allows increased design quality with a shorter time to market.

REFERENCES

[1] Networks on Chips: From Research to Products
Giovanni De Micheli, 2010, Association for Computing Machinery (ACM), DOI:10.1145/1837274.1837352
[2] A Flexible Multiprotocol Cache Coherent Network-on-Chip (NoC) for Heterogeneous SoCs, Michael Frank, Arteris, Linley Fall Conference, October 2020
[3] https://www.arteris.com/products/non-coherent-noc-ip/flexnoc
[4] https://www.arteris.com/products/coherent-noc-ip/ncore
[5] https://www.synopsys.com/verification/virtual-prototyping/platform-architect.html
[6] https://www.iese.fraunhofer.de/en/innovation_trends/autonomous-systems/memtonomy/DRAMSys.html
[7] Andrew B. Kahng, et al., "VLSI Physical Design: From Graph Partitioning to Timing Closure," Springer (2011)
[8] Performance Analysis for Network-on-Chip Development Using Synopsys Platform Architect, Synopsys Virtual Prototyping Day 2023