

Playing Well with Others: How NoC Compositions Enable Global Team Design

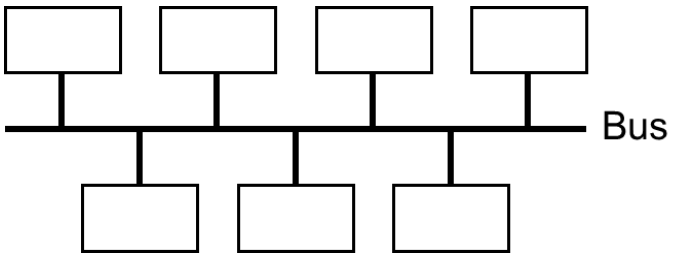
Jonah Probell, Arteris

Abstract— Chips are assembled of subsystems that are designed by different teams. NoC compositions, connected by the NoC Socket Protocol (NSP) are a valuable enabler of team design. Many technology and workflow considerations have been incorporated into the definition of the NoC composition standard.

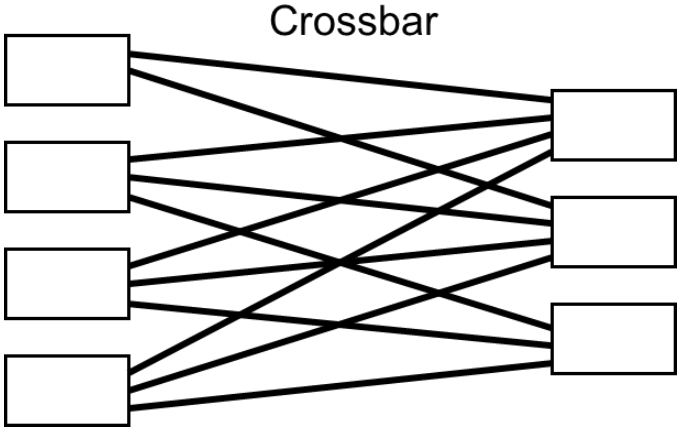
I. The Advent of Network-on-Chip (NoC)

A revolution took place in the mid-1990s. Chips started being designed as assemblies of IPs. Each IP was designed by a team. Companies had multiple IP development teams, and third-party licensing companies such as ARM and Imagination Technologies formed to offer standard IP. Bus interface standards were developed to support interoperable integration. AMBA is the family of bus interface standards that has been most successful.

Bus interfaces support the notion of transactions. Transactions are reads or writes of data associated with an address. In read transactions the data of an address is transferred from a slave to a master and in write transaction data of an address is transferred from a master to a slave.

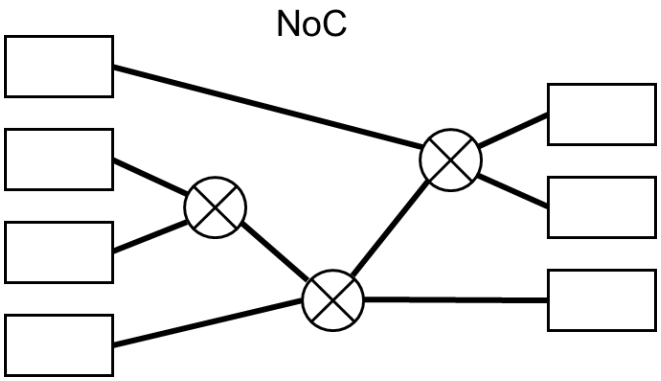


Transactions were, at first, performed on common busses shared by multiple IPs. As newer systems required great data transfer throughput, crossbar switches were developed. These provided greater throughput by allowing transactions between different pairs of masters and slaves to happen concurrently. As further integration of more IPs occurred, large crossbars became difficult to physically place and route. Network-on-chip technology was developed to provide the concurrency of crossbars in a way that is more amenable to physical design of chips.



A network-on-chip facilitates transactions between initiator and target IPs by sending packets. Packets comprise a header and sometimes a payload that transport either a request or a response. A read request packet transports an address from an initiator IP to a target IP. A read response packet transports data from a target IP to an initiator IP. A write request packet transports an address and data from an initiator IP to a target IP. Write response

packets signal the completion of a transaction.



II. Subsystems

To put ever-greater numbers of ever-smaller transistors to work in chips requires integrating ever-more IPs. Within the past few years, a new way of designing chips has taken broad hold of the industry. Subsystems, such as CPU clusters, memory complexes, graphics, and multimedia subsystems are assembled from IPs and chips are designed as assemblies of subsystems, each designed by a different team. Teams are increasingly located in different offices, often located across the globe.

To create families of chips that serve different market segments one subsystem can be replaced with another that meets a different specification. Though it would be ideal to optimize a single NoC for all IPs in a chip, it is impractical. Instead, each subsystem has its own optimized NoC. Unfortunately, each revision of each subsystem affects the entire SoC. This requires frequent revisions to software-visible address maps and top-level functional and performance verification.

NoCs are generated by a configuration tool such that each NoC has a specific packet header format and routing tables optimized for its topology. Since the packet formatting is different for every NoC, subsystems are connected through standard transaction interfaces that use topology-independent addresses. This requires decoding packets in an upstream NoC and encoding a packet in a downstream NoC. That adds significant, though unavoidable, latency to transactions.

One approach to connecting subsystems would be to create a top-level NoC that connects all subsystems. This is nice for implementing a simple hierarchical address map, verification environment, and performance simulation. However, using a top-level NoC is sub-optimal because it requires two steps of decoding and encoding packets between subsystems. For latency-critical paths, such as between a CPU cluster and a memory subsystem, the latency would be harmful to performance.

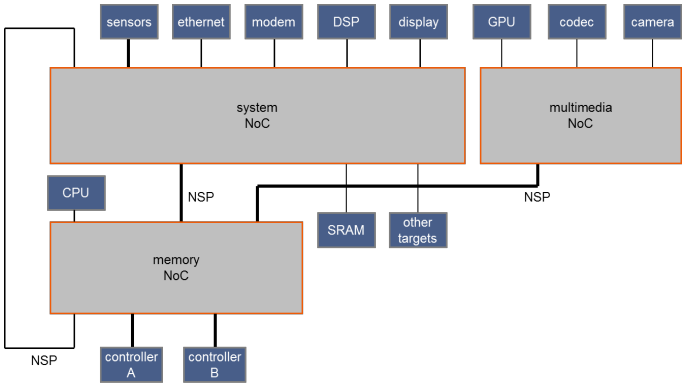
A superior approach is to directly connect subsystems as necessary for the desired functionality. This has complications, including:

- Different IPs see different address maps
- There is no single top-level NoC to model the full chip connectivity
- There is no single top-level NoC to simulate for performance analysis

These complications are addresses by NoC Compositions.

III. NoC Compositions

An Arteris NoC composition is a design object that encompasses a number of NoCs and defines their shared socket and port connections. The FlexNoC software uses the connectivity in a NoC composition and the address maps of its constituent NoCs to derive the connectivity and address map of each target in the system as seen from the view of each initiator for each mode of operation. While a NoC with separate request and response networks are naturally deadlock free, a NoC composition allows FlexNoC to check for routing loops between NoCs that could allow for deadlocks.

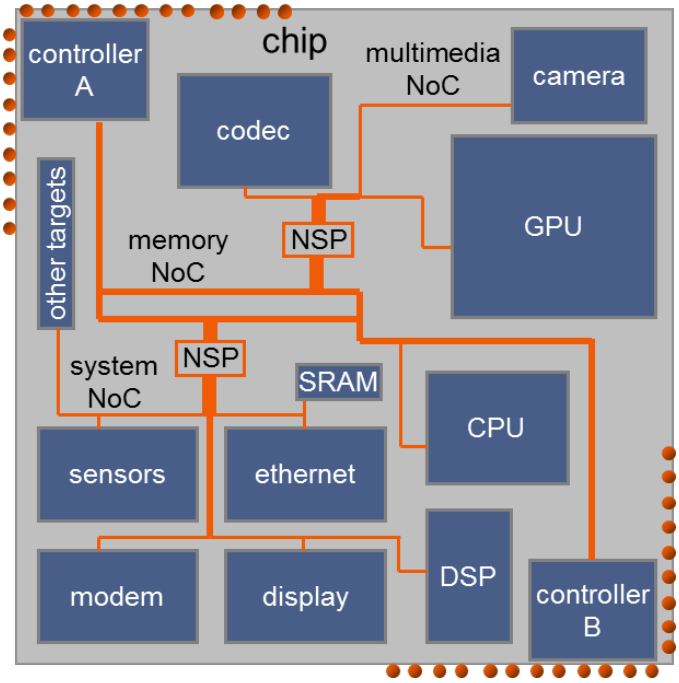


FlexNoC also uses NoC compositions to build top-level models of all interconnect logic. This allows the automatic generation of a full interconnect simulation model, which in turn is useful for a full system-on-chip simulation. Such

models formatted in RTL allow for functional verification. Formatted in SystemC they allow for performance verification.

IV. NoC Socket Protocol (NSP)

No particular interface protocol between NoCs is mandated by a NoC composition. AMBA, and other widely used transaction interface standards, provide the benefit of interoperability between independent subsystems. However, aspects of the AMBA protocol require logic delay that is unnecessary for a connection between like-minded NoCs. The Arteris NoC socket protocol (NSP) simplifies the interface between NoCs and thereby reduces latency.



Physically, each subsystem can be integrated into one complete full chip-level interconnect fabric without requiring bridges. This means that NoC compositions are applicable to fully abutted and channeled floor planning methodologies.

V. Benefits and Conclusion

Automating the integration of the NoCs of different subsystems reduces the limitations on the practicality of parallel subsystem design methodologies. This improves the granularity with which SoCs designs can be partitioned, and thereby the ability to delegate design work to specialists in different geographic locations across the globe. Finer granularity makes constituent projects simpler, which decreases bug rates and reduces development time.

Jonah Probell is a Senior Solutions Architect at Arteris. He has worked in the semiconductor intellectual property field for 15 years as an engineer, entrepreneur, and writer. His expertise is in processor, digital video, and network-on-chip design. As chair of the Synopsys Users Group of Silicon Valley, Jonah is interested in tools and methodologies for successful team design. He has written dozens of papers and patent applications.