

Survey on Real-Time Network-on-Chip Architectures

Keywords:

scalable, predictability, packet loss, latency, jitter, throughput, QoS: BE and GS, bandwidth, SDM, lane division multiplexing, TDM, looping containers, virtual circuits

Abstract

- i. Real time applications make out a significant portion of the embedded field.
 - ii. NoC *imposes complex constraint* on overall system design.
 - iii. Areas are QoS support, fault tolerance and adaptivity.
-

1. Intro

- i. Recent developments in nanotech allowed the integration of multiple processing elements (**PE's**) onto a single chip -> **MPSoC**.
 - ii. NoC have been proposed as the *emerging solution* for the increasingly complicated communication requirements of MPSoCs and future Many-Cores. NoCs offer a **scalable and flexible** communication infrastructure with highly supported modularity and powerful performance.
 - iii. Merit of MPSoC lies in their massive computational power and unique energy efficiency in a compact design.
 - iv. **Constraint:** real-time communication performance relies on both the logical result as well as the completion time bound.
(This means both the computation and the communication between the components must complete within certain deadlines for the system to behave correctly.)
-

2. Real-Time Challenges in NoC Design

- i. Extra paramount constraint: **predictability**: Messages belonging to a real-time flow have a strict deadline denoting the maximum time by which they must reach their destination.
(*Note: For hard real-time systems : no deadline is allowed to be missed even under worst case communication scenarios, as it would cause a catastrophic consequence.*)
- ii. This means, transmission delays between the NoC routers must be predictable in order to be able to guarantee an end-to-end latency which in the worst case does not exceed the imposed deadline.
- iii. Predictability is challenging constraint because the medium is shared between several concurrent applications and congestion situations.
- iv. So, aim is to have **guaranteed services** in terms of bandwidth and latency.

- v. Other constraints are : reliability and energy efficiency.
- vi. Continuous shrinking of silicon tech puts reliability and fault tolerance vulnerable.
- vii. Several conventional techniques to minimize NoC power consumption affect the timing behavior of the NoC.
- viii. For efficient execution of real-time applications on NoC-based MPSoCs is governed by a powerful task scheduling and mapping technique on the available cores.
- ix. Use of adaptive techniques is justified as real-time MPSoC's are subject to varying workloads and application-specific demands.

3. NoC Arch for QoS Support

Def: Quality of service denotes the ability of the network to provide the communication requirements and abide to the constraints imposed by the executed application.

Parameters: packet loss, latency, jitter and throughput

2 categories:

BE : best effort services (correctness and completion of transmission)

GS: guranteed services (real-time throughput and latency QoS guarantees)

- i. According to Mello, to support QoS in NoC designs, several approaches:
 - i. tailoring the network dimesions to fit the bandwidth requirements of all executed tasks in the system.
 - ii. supporting connection-oriented communication.
 - iii. implementing connection-less communication with priority scheduling.
 - iv. hybrid combination of above

3.1 Connection-oriented approach

- i. Idea is based on **resource reservation**. communication links from the source to the final destination of the data are reserved establishing a complete connection path for the packets before being injected into the network.
- ii. Implemented using : pure circuit switching or virtual circuits.
- iii. **Pure Circuit Switching:** communication is achieved over three steps:
 - i. path setup phase to check for free route and reserve the corresponding links.
 - ii. actual data transmission.
 - iii. cancellaetion pahse to release the resources.
- iv. So, for hard real-time applications, this approach has indisputable advantage in terms of reliable QoS support providing hard throughput guarantees and tight latency bounds.
- v. With an established route is an ownership for the traffic source, routers complexity is reduced and data is transferred at full rate of the links with no contention.
- vi. **latency:** it becomes only a factor of the route length and the dependency between traffics is confined to the route setup phase which fails or succeeds based on the traffic situation.
- vii. **drawbacks:** non-scalability, path setup delay and inefficient resource utilization due to wasted

resources to cover worst case behavior.

viii. SoCBUS: first pure circuit switched NoC:

- i. mesochronous clocking and a new packet based setup scheme.
- ii. to establish communication path: special request packets are sent following a distributed dynamic minimum path routing.
- iii. a req reaching its dest *locks* all the temporary reserved links travelled by the special packet.
- iv. a blocked request route at any node initiates a *retry phase* by retracting the partially established path and waiting for the occupied channels to become available.

ix. SDM: Spatial division multiplexing:

- i. works in a synchronous circuit switching NoC.

x. LDM: Lane division multiplexing:

- i. allows physical separation between data streams.
- ii. links are divided into lanes and each lane can be used for separate physical connection.
- iii. a connection may use one or more lanes based on bandwidth requirements.

xi. Pham proposes:

- i. distributed dynamic path setup scheme based on backtracking probing to avoid blocked channels.
- ii. Instead of waiting for busy channels to become available, the setup headers are allowed to flexibly backtrack and seek minimum path alternatives.

xii. **Nostrum and AEthereal** : two NoC examples based on TDM for better resource utilization while maintaining the hard performance guarantees.

- i. TDM - alternative to pure circuit switching which allows links to be shared in a contention-free scheme by reserving resources for specific points in time.
- ii. Data are injected into the network at dedicated time slots following a periodic schedule.
- iii. TDM schedules are either static or dynamically computed and they may be placed locally at each router for distributed routing or globally in the network interfaces (NIs) only for source routing.
- iv. **Nostrum** supports guaranteed service by implementation of **virtual circuits** based on TDM, where Virtual ckt's are set up semi-statically with fixed routes at design time however with variable bandwidth in runtime using the concept of **looping containers**.
- v. A special type of packets called container are loaded with information and are looped between the source and destination.
- vi. **AEthereal** supports both BE and GS traffic with static distributed TDM slot tables. Here, the TDM approach is implemented based on global synchronicity where all routers must always be in the same time slot.

xiii. Check the table1 for comparisons based on routing, clocking and traffic support for various implementations of NoC.

3.2 Connection-Less approach

Here, routers work in an independent manner.

- i. Involves packet switching with priorities.
- ii. Typically used of NoC based MPSoCs in multifunctional systems where diverse applications run in parallel bidding for different service requirements *real-time or non-real-time*.
- iii. *Traffic flows* are **clustered** according to their QoS requirements into different groups each assigned a priority value.
- iv. In contention situations, higher priority flows are serviced first and have the right to interrupt the lower priority ones.
- v. Wormhole switching with virtual channels is typically used for its high throughput and low buffering requirements.
- vi. Additionally, the prioritized traffic flows are highly applicable for a virtual channel flow control router, where each VC has a particular priority that corresponds to one of the traffic clusters.
- vii. **Advantages** : benefits in terms of bandwidth and network resources utilization as well as adaptation to the varying network traffic.
- viii. **Challenges**: in terms of complexity and latency problems in addition to weak QoS support due to sharing of resources which is always *critical* regarding *predictability*.
- ix. **Comparison**: 3.1 vs 3.2

- i. for variable bit-rate applications, connection-less approach proves better performance over connection-oriented approach in terms of guaranteed end-to-end delay for individual flows with better prediction of transmission latency.
- ii. However, these guarantees are only attainable for higher priority traffics.
- iii. Progression of lower priority traffics through the network highly depends on the load of the higher priority flows possibly leading to long time stalls.
- iv. Above problems can be tackled by presenting efficient priority assignment techniques, priority arbitration techniques, and architectural solutions.

x. QNoC:

- i. it is a packet switching NoC supporting four different classes of communication services.
- ii. Each class has a fixed priority and additional service levels can be further defined by adhering priority ranking.
- iii. Packets with higher priority are serviced first and round robin arbitration is applied for contending packets within same service level.

xi. BiNoC:

- i. it supports GS traffic with a reduced end-to-end latency.
- ii. It is based on wormhole switching with prioritized VCs'.
- iii. However, it offers dynamically self-reconfigurable bidirectional communication channels based on the real-time traffic need providing better bandwidth utilization.
- iv. Like, Idle channels can temporarily change their direction of operation to resolve congestions on the opposite direction.

xii. There is some research done (check paper) to address the problem of blocked lower priority data. 2 approaches:

- i. First, an aging mechanism which tackles the starvation problem of blocked data; the priority of the blocked data is incremented after certain number of cycles.
- ii. discarding mechanism which improves the bandwidth utilization by dropping packets of lower priority data if their deadlines are missed and if the number of dropped packets can be tolerated within the soft real-time application.

xiii. PRNoC: Predictable Reconfigurable NoC:

- i. proposes a round based priority arbitration scheme.
- ii. the term round refers to the set of packets arriving simultaneously when a new arbitration is initiated.
- iii. The proposed arbitration guarantees QoS for higher priority packets while avoiding starvation for lower priority ones.
- iv. Here Network Interface (NI) component supports PE(processing element) reconfiguration.
- v. Furthermore, switches support both deterministic XY routing as well as the odd-even turn adaptive routing model.

xiv. Last para discusses about techniques like GALS NoC, quota setting etc.

3.3 Hybrid Techniques

- i. connection-oriented approach can provide hard performance guarantees for GS traffics while connection-less approach offers soft guarantees with better adaptation to the varying network traffic as well as better bandwidth and resource utilization.
- ii. real-time applications, mainly targeting MPSoCs, invoke a **mixture** of traffic types with different predictability and QoS requirements. So, hybrid NoC designs are promoted to provide the right mix of soft and hard real-time guarantees and to efficiently handle both streaming and best effort traffics generated by realtime applications.

iii. MANGO NoC: typical asynchronous hybrid architecture

- i. routers internally consist of a BE router and a GS router implemented separately.
- ii. The BE router supports simple source-routing for connection-less data packets.
- iii. GS router utilizes (VCs) routing to establish hard GS connections

iv. Lusala proposes a hybrid real time NoC :

- i. it consists of two sub-networks: circuit- and packet switching.
- ii. The circuit switching sub-network is based on a combination between the SDM and TDM techniques in an attempt to provide a guaranteed service with increased path diversity and improved resource utilization.
- iii. This significantly improves performance.
- iv. Results are shown in table3 of paper.