

RESEARCH

Open Access



Congestion aware low power on chip protocols with network on chip with cloud security

Suresh Ponnan^{1*}, Tikireddi Aditya Kumar¹, Hemakumar VS¹, Sakthieswaran Natarajan² and Mohd Asif Shah³

Abstract

This article is to analyze the bottleneck problems of NoC in many more applications like multi-processor communication, computer architectures, and network interface processors. This paper aims to research the advantages and disadvantages of low congestion protocols on highway environments like multiple master multiple slave interconnections. A long-term evolution and effective on-chip connectivity solution for secured, congestion aware and low power architecture is emerged for Network-on-Chip (NoC) for MCSoc. Applications running simultaneously on a different chip are often exchanged dynamically on the chip network. Of-course, in general on chip communication, resources mean that applications may interact with shared resources to influence each other's time characteristics.

Keywords: Network-on-Chip, System on Chip, Chip multiprocessor, Congestion, Long term evolution, Embedded transition inversion, Very Large Scale Integration

Introduction

In order to further scale the overall performance within a limited power budget, future computing systems are expected to incorporate a large number of processing and storage components. Processors today often have 2 to 8 cores on a chip. Very advanced processors like Intel Single-Chip (SCC) and the Tilera family TILE64 already have several cores. There are no other cores. Such various core systems are probably going to depend on the network on the chip and are regularly called network-on-chip (NOC) for productive communication. Many core systems need to perform multiple working parallel loads and dynamically share resources in order to completely use a large number of processing elements. Cloud computing infrastructures, for example, can support multiple, physically shared virtual machines. Therefore, NOC systems applications may interfere in the execution

of each other via network contention and there may be contention and delay in the application communication traffic, that would not have happened if every application had been executed exclusively. The network conflict over shared waterways obviously is a presentation challenge and furthermore presents reasonableness and service quality issues as of late analyzed. In this paper we examine the wellbeing implications of the mutual on-chip network, specifically information spill through interference with the network, and propose a productive protection mechanism. In general, network interference latencies and transuded variations can be utilized by an aggressor as timing channels either to find confidential information from a high-security ensured framework (second-chain attacks) or to intentionally subtly erase information from a noxious program when direct channels are protected (covert-chain attacks). During the investigation of a likely denial of service (DoS) assault on the network, this paper mirrors the primary exploration on time channels, to the best of our aptitude. For frameworks that require significant levels of unwavering quality, the board of both side channel and side channel issues are significantly

*Correspondence: ohaasif@kdu.edu.et

¹ Department of ECE, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, Chennai, India
Full list of author information is available at the end of the article

analyzed. Consider, for instance, distributed computing foundations which license physical equipment on virtual machines from a few clients. To be attainable for organization or military customers, a program must be confident that crucial business mysteries cannot be overlooked. Nonetheless, the present contracts for distributed computing services that disallow a service supplier from sharing physical frameworks between a few clients to address these concerns. In this manner, the wellbeing basic frameworks have been modelled and analysed, for example, a motor control except if there is a decent assurance of insulation, to fulfill time-frames, cannot exploit multi-core frameworks for multiple undertakings.

Network on chip communication

A chip network is designed to allow messages to flow from the source module to the target module through various links involving switch routing decisions. It has multiple data connections, linked by switches from point to point. It can be defined as a structured, scalable fabric network.

Features of NOC

1. Chip network helps to simplify the hardware necessary to route and switch functions.
2. Support for several network areas is available in the multi-topologies and multi-options sectors.
3. In conjunction with the network on a chip, scalability, interoperability and application creation are improved.
4. Relative to other architectures, network on a chip improves the power efficiency of complex system-on-chips.
5. Synchronization issues are better handled than other projects. In most system-on-chips, the wire routing interference is also better handled by a network.
6. Chip network provides higher frequencies for service.
7. It's far easier to implement timing closure.
8. Checking problems through their well-designed and structured approach is much simpler.

Literature review

Hu et al. [1] implemented a buffer sizing approach for mid-node NoCs using formalisms in queuing theory. The principal aim is to decrease the mean delay of all NoC communications with decreased occupancy of the buffer region. The authors saw data storage as an atomic unit, i.e. mode for store and forward switching. The store-and-forward technique is not popular in NoCs, because the

NoC buffer size should be as small as the maximum size packet, which will increase the latency and the area. The authors use a synthetic model from Poisson to describe telecommunication networks in the field of transport modelling. In contrast to models on the basis of trace or related ones, the inconvince of this model is decreased accuracy. The self-similar feature of mpeg traffic, a standard application in existing socs, was demonstrated by Varatkar and Marculescu [2]. In order to avoid buffer overflow, the authors have proved that the optimal buffer size of MPEG decoder modules can be specified. In terms of the modeling of traffic within the chip, a synthetic method for processing traffic is provided in order to combine traces of traffic and its statistics using a synthetic track process. Experimental traffic models only considered point-to-point interactions and ignored the potential effect of concurrent flows. Further information on the buffer threshold value is not given by the authors. The method for buffer size considering data generation and consumption levels for packets transmitted in the burst was proposed by Chandra et al. [3]. The result is to compare atomic and distributed buffers by using the performance metric. Atomic buffers are in target IPs, whereas the buffers on distributed intermediate nodes are placed. There is an improved contribution to the distributed buffer strategy for results reported in [3]. The key drawback of this approach is that the NoC is designed for a fixed traffic scenario that is unsatisfactory for SoCs that embrace post-design applications. In light of latency constraints and packet drop estimates, Manolache and al [4]. suggest a framework focused on traffic heuristic measures for buffer size and occupational optimization. In this method, communication events on the network are mapped and/or packets are delayed at source, in order to avoid rivalry between different flows. To verify this approach the authors use a variety of experimental uses in experiments. The results show the impact of traffic shaping on intermediate routers, which reduces the total amount of buffer space available. As a result, the number of applications to be deployed within a single buffer space has increased. However, there are no assurances for this job. In compliance with traffic requirements, Nicopoulos et al. [5] provides a single buffer structure that assigns dynamically virtual and buffer resources. The authors use fixed (self-similar) injection rates for the experiments. The distribution of space traffic is also special. The results of this research underline a reduction of latency and an increase in output. However, guarantees on the efficiency and latency values of target routers are not included in the process. Coenen et al. [6] are using virtual channels and credit-based flow control in NoC, with a size buffer algorithm for an aim IP. The goal is to achieve a consistent consumption rate without

data loss for target IPs. The author uses two arrays which store information expressing the time the data arrival was provided and the data processing rate needed for an IP target for both dataset manufacturing and buffer consumption. We do not however consider rivalry between various flows, which would possibly alter the goal traffic aperiod in time. The routing algorithms provided in [5, 6] include a minimal algorithm for 2D mesh, known as "doble-y." It requires 2 virtual Y-dimensional channels and one virtual X-dimensional channel. In [7] the authors suggested a maximum adaptive double-y routing algorithm (Mad-y) to improve adaptiveness, maximizing the use of available resources (virtual channels) in comparison with the dual-y network algorithms [5, 6]. Li et al. Ming et al. [8] DyXY has implemented dynamic routing algorithm for storage-aware, which decides output channel based on storage status for neighboring nodes' buffers. The congestion-conscious complete adaptive routing algorithms, RCA [9], DBAR [10] and CATRA [11], use non-local data congestion to route the packet using extra hardware. Nevertheless, the majority of NoC's architectures are powerful [12] and lack special hardware to guarantee QoS. There are now very few works that measure the worst bandwidth and delay values for a BE NoC as far as we know it best. Balakrishnan and Ozguner [13] suggests the lumped connection model, where the links of a packet are connected to a single connection. The model does not differentiate direct conflict (by lack of arbitration). The calculated limits are negative regardless of complete buffers along the way. In [14], the Qian et al. methods for evaluating real-time boundaries for NoCs based on network calculus [14, 15], which use service curvature and arrival curve characterizing the operation characteristics of switches and injected traffic. For many applications, obtaining arrival curves is not an easy task. Traffic control may therefore be needed to ensure that the amount of Traffic injected does not exceed a defined level for a given period of time for an arbitrary injected traffic load. In [16] the problem of buffer optimization is overcome in the worst case of network calculation efficiency limitations. In [17] Bakhouya et al. also present a network calculus-based model to estimate the maximum size to end-of-end and buffer for mesh networks; the delay limits for the flows are not strict and actual values can be greater. Hierarchical solutions to various power management and failure tolerance control systems. The tradeoffs between field, energy and latency overhead motivate and motivate separate dedicated intermediate communication monitoring networks [17–19]. In [20] a conscious deflection routing algorithm (Fault-on-Neighbor, FoN) is proposed for NoC, which allows routing decisions on the basis of neighbor switch link status in 2 hops to avoid defect connections and switches. A faulty

adaptive deflection routing algorithm was proposed in [21] which makes cost-based routing decisions. Both incoming packets are priority based on the number of hops that the packet was routed. The most significant goal is the packet with the highest hop count. For each packet, the switch makes routing decisions from top to lowest priority [22]. The device configuration involves resource usage and hierarchically arranged power supplies. The approach to design of circuits VLSI is consistently focused on unforeseeable discrepancies and extreme power constraints [23]. By adopting bio-inspired architecture of the framework, the hierarchic agent is proposed to track the NoC design process. It has hierarchical power surveillance, in which various agent levels work together to lower contact power HAM offers a systemic approach that uses the hierarchical monitoring structure [24]. In order to ensure compatibility, even if network portion is out of operation due to defects, the structural redundancy of the chip network is used with adaptive routing algorithms [25]. The distributed method of fault diagnosis makes it easier to assess the fault status of NoC switches and their connections. A static XY routing algorithm for two dimensional networks was developed, with static XY routing initially crossing the packet in X dimension and then crossing into Y direction. A static XY routing algorithm was proposed to help the design of the adaptive and not adaptive routing algorithms for various network architectures. This algorithm is free from dead locks, but adaptivity [26] is supported. A NoC is an infrastructure for on-chip communications that implements multi-hop communication, primarily packet-based. NoC can use the communications tools more effectively than conventional on-chip buses via pipeline packet transmission. Generic NoC structures decrease the difficulty of the VLSI system in contrast to custom routed wires. Development in technology has also led researchers to rethink their opinions on NoCs. In addition, some of these works concentrated on 3D NoC [27]. Delivering 3D NoC modeling and simulation tools [28], developers have also expected this advance. The goal is often to minimize coastal areas and electricity consumption without reducing system performance as far as production and latency are concerned [29]. The word ReNoCs, meaning Reconfigurable NoCs, is increasingly being developed within the scientific community, which resulted in some initiatives on this topic being developed [30]. With this basic socket, a distinctive prerequisite is that unique interface instantiations should be made available in a variety of measurements (bus distance, device handshaking). The Open Core Protocol (OCP) [31] is a growing socket. The OCP specification describes a versatile family of core centric memory-matched protocols for on-chip systems as a native core

interface. Certain suggested specifications were incorporated into the NA architecture, such as Virtual Component Interface (VCI) [32], which were used in the NoCs SPIN [33] and Proteo [34], Advanced extensible Interface (AXI) [35], System Transaction Level (DTL) [36], and Wishbone [37]. The CI actually implements the OSI layering model client layer services. The designers decided to convert the centralized, not scalable bus-based systems in a chip into a new distributed, low-power, scalable, secure, guaranteed operation, package-based and internet-based protocol-based layered networks, called Chip-based network (NoC) and deal with issues such as over-order transactions, higher latencies and end-to-end flow-control [38–42]. Including a collection of routers (r), links (l), intellectual property cores and network adapters (NA) [43, 44] are part of the NoC, which offers both parallel and multi-core computing platform. The routers are connected with point-to-point links due to the fact that a router can be clustered through NA [45–47], with more standardized or heterogeneous Processing Element (PE) known as IP center. The NA is regarded as a single hardware entity that unites computation and communication and enables the reuse of IP core and communication infrastructures [48, 49]. Research in NoC is divided into four areas: (1) device, (2) adapter network, (3) network and (4) connection [50]. Table 1, showing the link between these research areas, the basic components of NoC and the layers of OSI, indicates the network data flow [51]. Recent work shows that adaptive channel buffers (storage on the connection), by minimizing or removing hungry power buffers, can significantly reduce power consumption and overhead area [51]. The following results: This addresses the drawbacks of buffers and therefore adds less buffer [52]. The router, which affects data transmission latency, chip area and power consumption, is the main component of a network-on-chip system. Safety in NoCs was studied from several angles focusing on specific attenuation mitigation such as denial-of-service (DoS) defenses, battery drainage attacks

[52], and access control in shared memory systems of different regions [53, 54], and buffer overflow attacks [55, 56]. Gebotys and Zhang concentrate on the continuity of data transmitted by the NoC in SoC settings by including encryption techniques [57].

In addition, the systematic method to monitoring flows has been suggested in a number of computer hierarchies from virtual machinery [58], high-level languages [59] and builders [60] through to binary analyzes [61], and even hardware-supported information flow tracking systems [9, 21, 62]. Approaches operating at language level can also control implicit flows due to branches and loops which allow non-determinism to be implemented in the program. Since code never executed can pass information through (by lack of execution), other safe languages delete undetermined conduct (whether entirely or on the basis of hidden or untrustworthy terms) from the program code [63]. At OS level, Flume [64] also has been shown to be secure via processes, pipes, file systems, etc., while seL4 [65] uses an automatic theorem which proves that the kernel has safety properties.

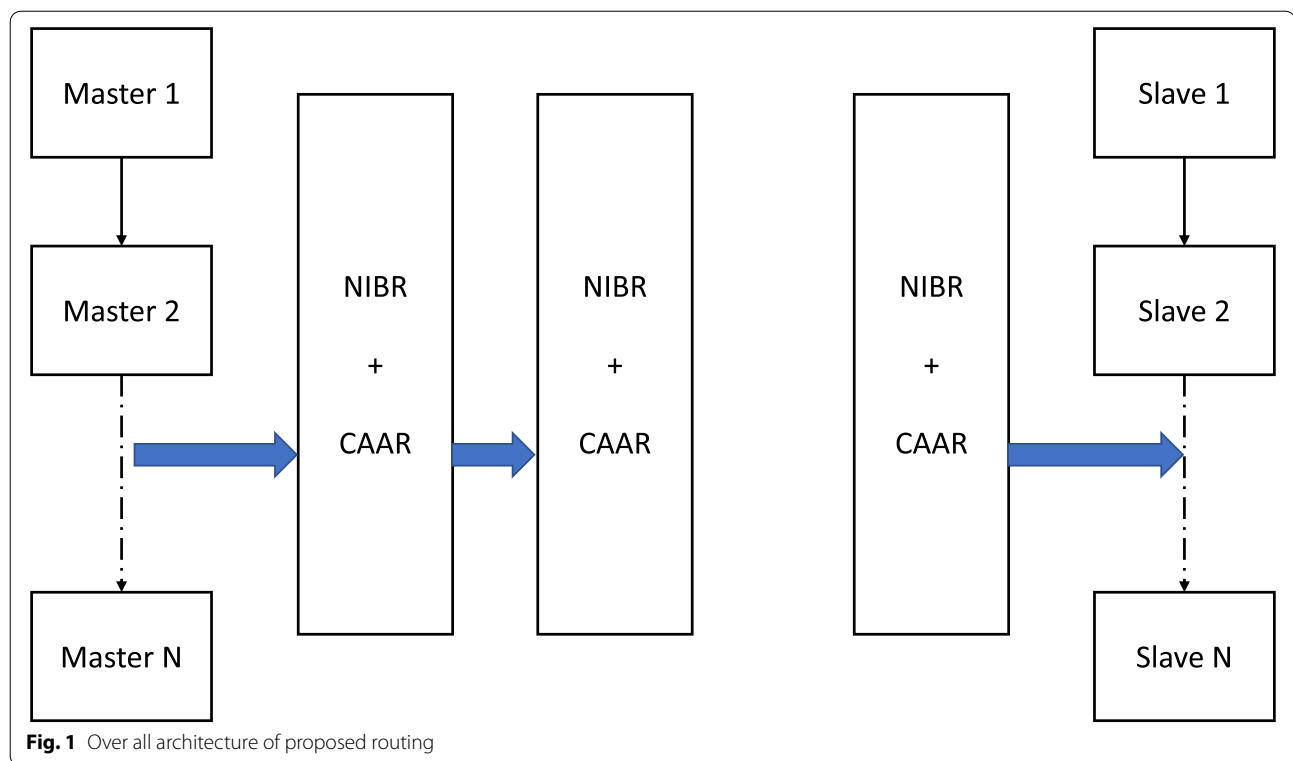
A regular separable allocator with dynamically assignable virtual channels, consisting of input arbiters and output arbiters [66]. The network conflict over shared channels obviously concerns efficiency and also poses fairness and service quality problems, which have recently been examined [67, 68]. Authors compare in depth the different component frameworks that are adapted for ETS and embedded systems in [69]. For example, the PCOS project [69, 70] defines an embedded system component model that is designed specifically for field devices. The CIAO project is our own related initiative to provide distributed, real-time and embedded systems with component middleware [71–78]. The OMG's Lightweight CORBA Component Model (LwCCM) specification is being introduced by the CIAO. As its default contact middleware, CIAO uses the TAO CORBA object request broker (ORB). CIAO also supports asynchronous messages and the OMG Data Distribution Service (DDS) through its ports with the recent standardization of connector devices.

Table 1 Network components and layers

OSI Layers	SoC/NoC Layers	SoC Components
Application	System	Cores
Presentation		
Session		
Transport	Interface	Network Adapters
Network	Network	Routers
Link	Link	Wires
Physical		

Proposed architecture

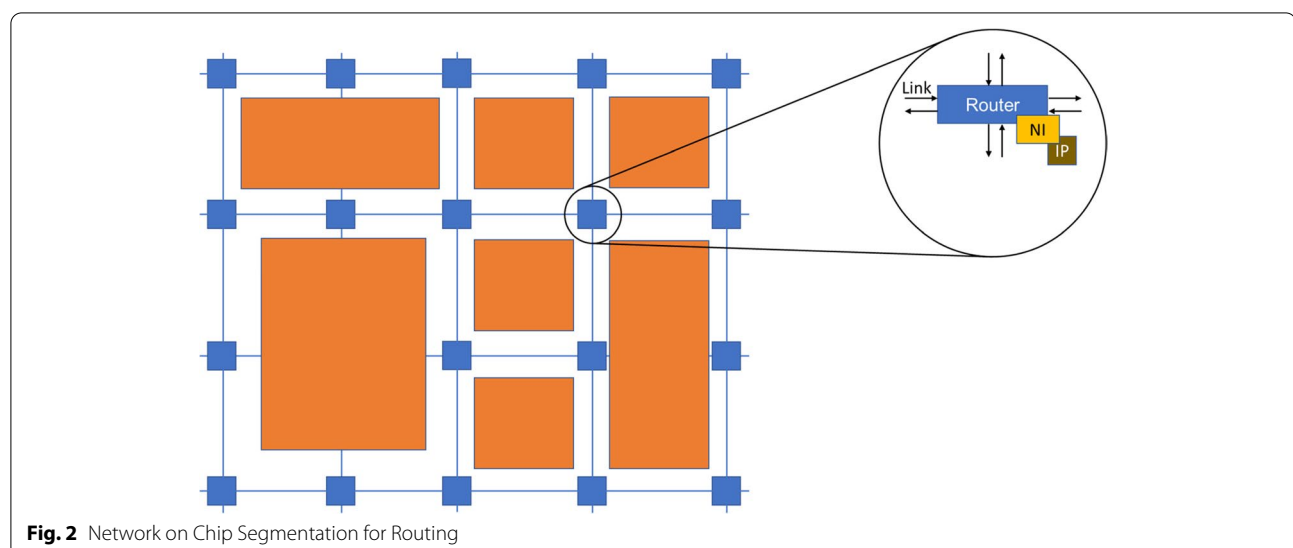
The Fig. 1 incorporates proficient solid and exact routing protocol utilizing low force data transfers. Congestion aware routing protocol, Non interface based versatile routing protocols are utilized for Efficient, precise and made sure about interchanges. Embedded transition inversion is presented for low power transmission.

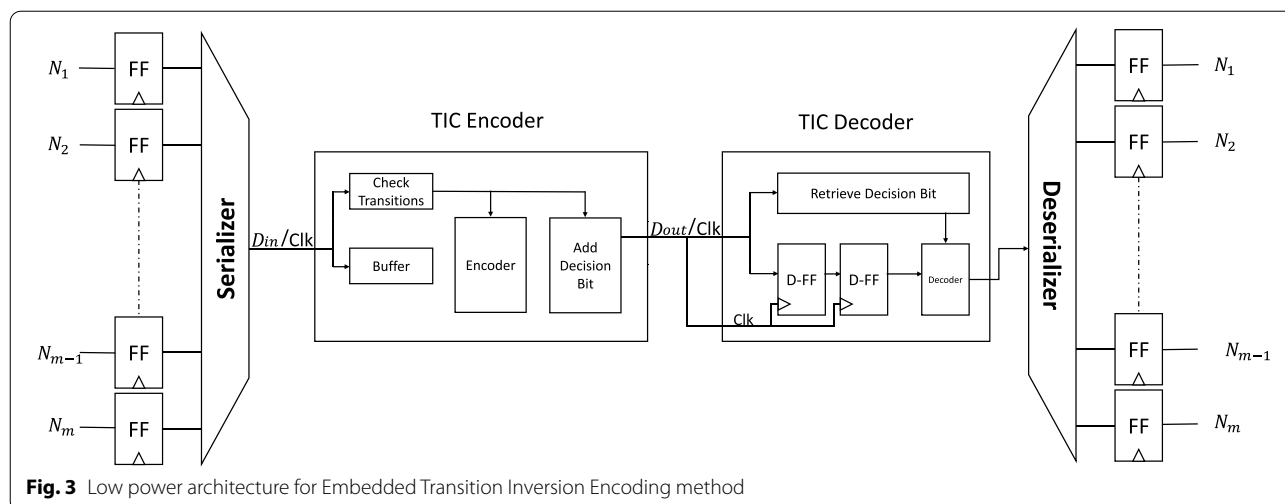


Network on chip segmentation

Here, number of nodes in each measurement is should be determined for segmentation purpose. Three or Four nodes ought to introduce in each segment. It is well established actuality that number of nodes more noteworthy than 5 thoughts to be again part in to areas for segmentation purpose other than three and four. The entire adding of routers ought not be equivalent. For

instance, twelve can be assembled up to multiple times of three or multiple times of four. Need to adjust the proposed calculation so that each yield of segmentation calculation ought to be one of a kind. As a well established truth the various yields have variable number of segments. Need should be given for the square which comprises a smaller number of segments is shown in Fig. 2. In like manner need to change the calculation rule and





we should see that each measurement should comprises three or four nodes. At long last, by doing these adjustments each yield of segment ought to be one of a kind to improve precision levels.

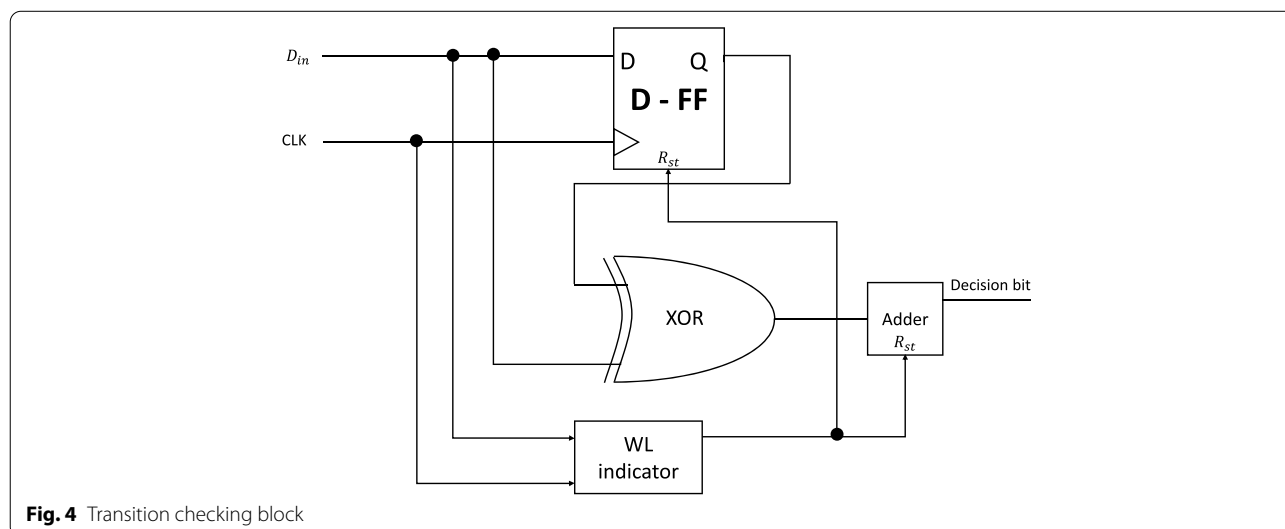
Routing algorithm

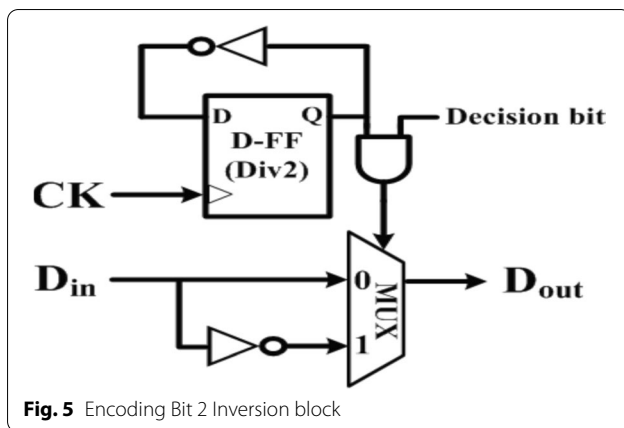
This proposed algorithm comprises absolutely three phases and the algorithm consider entire network in to various subnets. Segmentation will be done on entire network by accepting subnet cover based segmentation. First and for most stage follows the following divided subnet which is the closest nearby subnet dependent on steering algorithm. Among all these three stages, second stage is some troublesome procedure to pick limit hub as impermanent goal in that subnet respected on the yielded yield of first for most stage. Last collector of the procedure is should be assessed with this stage itself. At last, in the last third stage, bundle in its present subnet

will be sent to the un releasable goal which is determined toward the finish of past stage dependent on productive solid steering algorithm.

The entire procedure constantly run until its last goal comes to success as shown in Fig. 3. A good for nothing number is utilized here to speak to the congestion status of every hub, in light of that status last goal buffer will be filled by information by considering all past switch congestion status. The status of every individual buffer might be fall in four states dependent on worthless number inhabitation. Three static estimations of edges being considered in this no-good number state.

In above Embedded Transition Inversion (ETI) encoding procedure, referenced input port Data in is stored in ordinary gate level-based buffers. That activity holds up until the check transition idea is finished for process finish technique. Here, choice bit need to inferred for encoding reason. This choice can





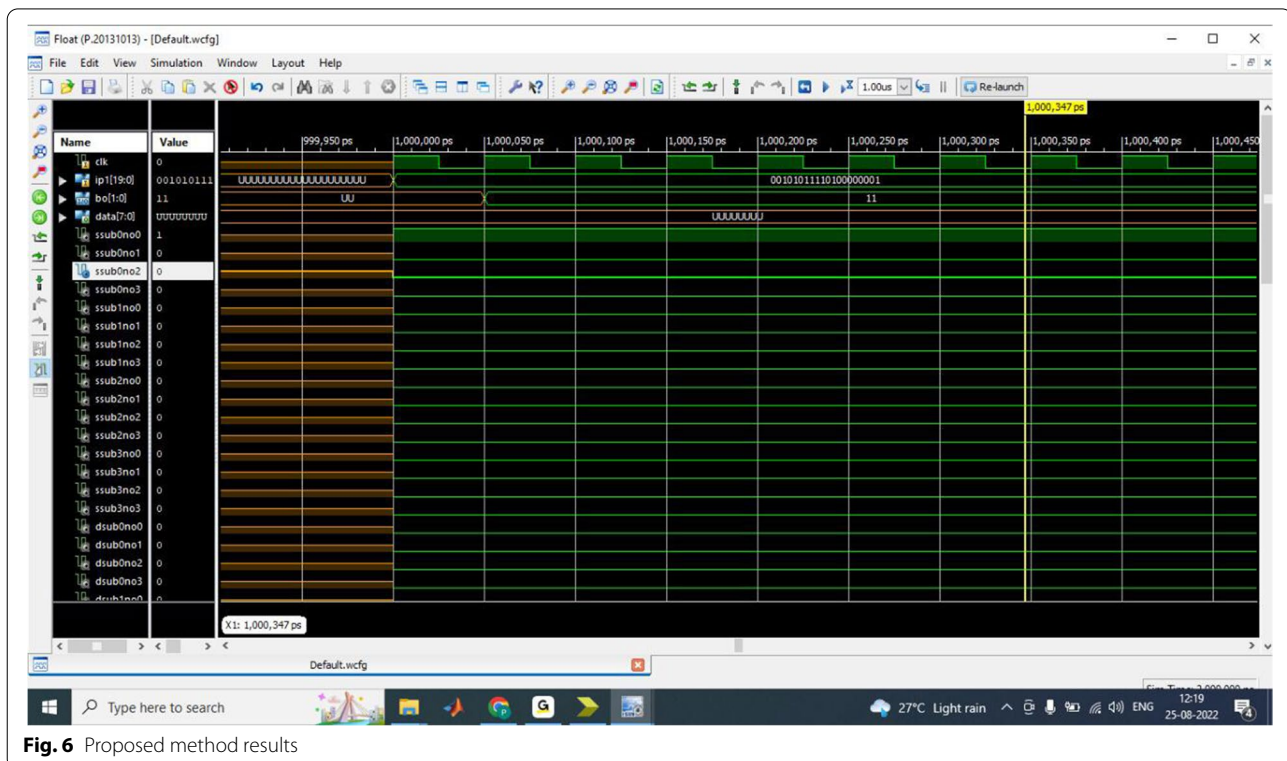
be created bases on number of transitions in a given word and edge an incentive for that data. Bit 2 inversion encoding calculation is performed dependent on yielded choice bit. In the event that choice bit is '1' infers, encoding should be performed. In any case encoding doesn't start any calculation. Relating stage encoding can be performed on inquiry if bit2 inversion is performed on input data.

As shown in above block in Fig. 4, if decision bit is '1' each and every second bit of given data is inverted, else same data will be processed. Phase encoding is also can be performed based on produced decision bit.

Coming to the receiver side decoder block, need to be detect whether data is encoded at transmitter side or not. This operation can be done by phase decoder at receiver side which will be available at first stage. This decoder finds whether any phase difference in data and received clocking signal. Here, at receiver side decision bit is again produced by finding the phase difference between data and clock. Decoding operation like bit 2 inversion will be performed on received data, if yielded decision bit is '1'. If yielded decision bit is '0' no decoding operation will perform on received data at receiver side block. At final stage, serial data will be converted in to parallelized data using de sterilizer which is combination of DE multiplexers. Fig. 5

Results

The clock frequency of the serial links is a lot higher than that of the equal transport. The higher clock frequency prompts issues, for example, buffering, clock synchronization, and plan intricacy. The Simulated ETI encoder and decoder accomplish great force putting something aside for working the serial link up to 2 GHz is shown in Fig. 6. High frequency circuit plan strategies are expected to defeat those issues while receiving the proposed ETI plan to higher frequency serial link. As we probably aware, serial link interface is regular received on off-chip interconnect, for example, USB and SATA.



Conclusion

This research suggested several different methods in order to describe NoC-based real-time SoCs bandwidth and latency, aiming at QoS guaranteed provisions. The choice of the most appropriate approach depends on the system's performance requirements and whether the NoC can provide unique hardware facilities or not. Those methods are intended for applications which require minimum latencies and injection regulation, while the other method is ideal for use where packet injection must be versatile for higher injected bandwidths on an average level and no hardware regulations can be made. Finally; latency, power optimizations without increasing density constraints and security improvement are main subjective improvement in almost all literature papers for quality-of-service NoCs.

Authors' contributions

Suresh Ponnan—Wrote the Manuscript. Tikireddi Aditya Kumar—Wrote the Manuscript. Hemakumar VS—Reviewed and given suggestion to the manuscript. Sakthieswaran Natarajan—Reviewed and given suggestion to the manuscript. The author(s) read and approved the final manuscript.

Funding

There is no funding support for the research work.

Availability of data and materials

Not applicable.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

None.

Author details

¹Department of ECE, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, Chennai, India. ²Department of Civil Engineering, EGS Pillai Engineering College, Nagapattinam, Tamilnadu, India. ³Kebri Dehar University, Kebri Dehar, Ethiopia.

Received: 23 June 2022 Accepted: 3 August 2022

Published online: 09 September 2022

References

- Hu J, Ogras UY, Marculescu R (2006) System-level buffer allocation for application-specific networks-on-chip router design. *IEEE Trans Comput-Aided Des Integr Circuits Syst* 25(12):2919–2933
- Varatkar G, Marculescu R (2004) On-chip traffic modeling and synthesis for MPEG-2 video applications. *IEEE Trans VLSI Syst* 12–1:108–119
- Chandra V, Xu A, Schmit H, Pileggi L (2004) An interconnect channel design methodology for high performance integrated circuits. In: *Proceedings of the Design, Automation and Test in Europe*. pp 1138–1143
- Manolache S, Eles P, Peng Z (2006) Buffer space optimization with communication synthesis and traffic shaping for NoCs. In: *Proceedings of the Design, Automation and Test in Europe*. pp 95–98
- Nicopoulos CA, Dongkook P, Jongman K, Vijaykrishnan N, Yousif MS, Das CR (2006) ViChar: a dynamic virtual channel regulator for network-on-chip routers. In: *39th Annual IEEE/ACM International Symposium on Microarchitecture, MICRO'06*. pp 333–346
- Chien A, Kim J (1992) Planar-adaptive routing: low-cost adaptive networks for multiprocessors. In: *Proceedings of 19th International Symposium on Computer Architecture*. pp 268–277
- Glass CJ, Ni LM (1992) Maximally fully adaptive routing in 2d meshes. In: *International Conference on Parallel Processing*, volume I. pp 101–104
- Li M, Zeng QA, Jone WB (2006) DyXY - a proximity congestion-aware deadlock-free dynamic routing method for network on chip. In: *Proceedings of 43rd Design Automation Conference*. pp 849–852
- Ramanujam R, Lin B (2010) Destination-based adaptive routing on 2D mesh networks. In: *Proceedings of 6th ACM/IEEE Symposium on Architectures for Networking and Communications Systems*. pp 1–12
- Ma S, Jerger N, Wang Z (2011) DBAR: An efficient routing algorithm to support multiple concurrent applications in networks-on-chip. In: *Proceedings of 38th International Symposium on Computer Architecture*. pp 413–424
- Ebrahimi M, Daneshmand M, Liljeberg P, Plosila J, Tenhunen H (2012) CATRA- congestion aware trapezoid-based routing algorithm for on-chip networks. In: *Proceedings of 15th Design, Automation and Test in Europe Conference Exhibition*. pp 320–325
- Salmine E, A Kulmala, Hamalainen T (2008) Survey of Network-on-Chip Proposals (www.ocpip.org)
- Balakrishnan S, Ozguner F (1998) A priority-driven flow control mechanism for real-time traffic in multiprocessor networks. *IEEE Trans Parallel Distrib Sys* 9(7):665–678
- Qian Y, Lu Z, Dou W (2010) Analysis of worst-case delay bounds for best-effort communication in wormhole networks on chip. *IEEE Trans Computer-Aided Design* 29(5):802–815
- Chang CS (2000) Performance guarantees in communication networks. Springer-Verlag, London, pp 1–392. <https://doi.org/10.1007/978-1-4471-0459-9>
- Le Boudec JY, Thiran P (2001) Network calculus: a theory of deterministic queuing systems for the internet. *Lecture Notes in Computer Science (LNCS)*, vol 2050. Springer-Verlag, London, pp 1–276. <https://doi.org/10.1007/3-540-45318-0>
- Jafari F, Lu Z, Jantsch A, Yaghmaee MH (2010) Optimal regulation of traffic flows in networks-on-chip. *Proc. Design, Automation and Test in Europe Conf. and Exhibition*. pp 1621–1624
- Bakhrouya M et al (2009) Analytical modeling and evaluation of on-chip interconnects using network calculus. *Proc. ACM/IEEE Int'l Symp. Networks-on-Chip (NOCS)*. pp 74–79
- Murali S et al (2006) Design of application-specific networks on chips with floorplan information. *Proc. IEEE/ACM Int'l Conf. Computer-Aided Design (ICCAD)*. pp 355–362
- Feng C, Lu Z, Jantsch A, Li J, Zhang M (2010) FoN: Fault-onNeighbor aware routing algorithm for Networks-on-Chip. *Proc. 23th IEEE Int. System-on-Chip Conf. (SOCC)*. pp 441–446
- Valinataj M, Mohammadi S, Safari S (2011) Fault-aware and reconfigurable routing algorithms for Networks-on-Chip. *IETE J Res* 57(3):215–223
- Valinataj M, Mohammadi S, Plosila J, Liljeberg P, Tenhunen H (2011) "A reconfigurable and adaptive routing method for fault-tolerant mesh based networks-on-chip", Elsevier. *Int J Electronics and Communications (AEU)* 65(7):630–640
- Yin AW et al (2008) Hierarchical agent monitoring NoCs: a design methodology with scalability and variability. *Proc. 26th NORCHIP Conf.* pp 202–207
- Guang L, Yang B, Plosila J, Latif K, Tenhunen H (2010) Hierarchical power monitoring on NoC - a case study for hierarchical agent monitoring design approach. *Proc. 28th NORCHIP Conf.*
- Guang L, Nigussie E, Rantala P, Isoaho J, Tenhunen H (2010) Hierarchical agent monitoring design approach towards selfaware parallel systems-on-chip. *ACM Trans Embed Comput Syst* 9(3):25

26. Kohler A, Schley G, Radetzki M (2010) Fault tolerant network on chip switching with graceful performance degradation. *IEEE Trans Comput-Aided Des Integr Circuits Syst* 29(6):883–96
27. Xie Y et al (2009) Three-Dimensional Network-on-Chip Architecture. In: *Three Dimensional Integrated Circuit Design*, Xie Y et al Eds., Springer US ed, pp. 189–217
28. Kahng A et al (2009) ORION 2.0: A Fast and Accurate NoC Power and Area Model for Early-Stage Design Space Exploration. In: *Proceedings of Design Automation and Test in Europe*
29. Zarkesh-Ha P et al (2010) Hybrid network on chip (HNoC): local buses with a global mesh architecture. In: *Proceedings of the ACM/IEEE international workshop on System level interconnect prediction*, pp 9–14
30. Krasteva YE et al (2010) Reconfigurable networks on chip: DRNoC architecture. *J Syst Archit* 56:293–302
31. Alliance O (2003) Open core protocol specification. In: Release
32. Alliance V (2001) Virtual component interface standard (<http://www.vsi.org/library/specs/summary.html>)
33. Guerrier P, Greiner A (2000) A generic architecture for on-chip packet-switched interconnections. In: *Proceedings of the conference on Design, automation and test in Europe*, ACM, pp 250–256
34. Ahonen T, Sigu"enza-Tortosa DA, Bin H, Nurmi J (2004) Topology optimization for application-specific networks-onchip. In: *Proceedings of the 2004 international workshop on System level interconnect prediction*, ACM, pp 53–60
35. ARM A (2004) AXI Protocol Specification, version 1.0 www.arm.com, ARM
36. Wang C, Chao K, Sivaperumal S, Suresh P (2020) "Anti-PVT-Variation Low-Power Time To Digital Converter Design Using 90 nm CMOS Process" *IEEE Trans Very Large Scale Integr VLSI Syst*;28(9):2069–2073
37. Opencores S (2002) Wishbone system-on-chip (soc) interconnection architecture for portable ip cores. http://cdn.opencores.org/downloads/wbspec_b3.pdf
38. Bjerregaard T, Mahadevan S (2006) A survey of research and practices of network-on-chip. *ACM Comput Surv (CSUR)* 38(1):1
39. Suresh P (2017) Creation of optical chain in the focal region of high NA lens of tightly focused higher order Gaussian beam. *J Opt* 46:225–230. Springer
40. Celestine I, Suresh P, Revathi M, Kathiravan S, Chaun Yu Chang (2019) An efficient and unique TF/IDF algorithmic model-based data analysis for handling applications with big data streaming. *MDPI – Electronics* 8(11):1331
41. Henkel J, Wolf W, Chakradhar S (2004) On-chip networks: A scalable, communication-centric embedded system design paradigm. In: *Proceedings 17th International Conference on VLSI Design*, IEEE, pp 845–851
42. Ramanujam RS, Soteriou V, Lin B, Peh LS (2010) Design of a high-throughput distributed shared-buffer NoC router. In: *Proceeding of the 2010 Fourth ACM/IEEE International Symposium on Networks-on-Chip (NOCS)*, IEEE, pp 69–78
43. Chan CH, Tsai KL, Lai F, Tsai SH (2011) A priority based output arbiter for NoC router. In: *Proceedings of the Circuits and Systems (ISCAS) on IEEE International Symposium*, IEEE, pp 1928–1931
44. Saponara S, Vitullo F, Petri E, Fanucci L, Coppola M, Locatelli R (2011) Coverage-driven verification of HDL IP cores. In: *Conti M (ed) Solutions on embedded systems*. Springer, New York, pp 105–119
45. Suresh P, Rajesh KB, Sivasubramonia Pillai TV, Jaroszewicz Z (2014) Effect of annular obstruction and numerical aperture in the focal region of high NA objective lens. *Opt Commun* 318:137–141
46. Bertozzi D (2006) Network interface architecture and design issues. *Networks on Chips: Technology and Tools*, The Morgan Kaufmann Series in Systems on Silicon, pp 147–202
47. Zimmermann H (1980) OSI reference model–The ISO model of architecture for open systems interconnection. *IEEE Trans Commun* 28(4):425–432
48. DiTomaso D, Morris R, Kodi AK (2013) Extending the energy efficiency and performance with channel buffers, crossbars, and topology analysis for network-on-chips. *IEEE Trans Very Large Scale Integr VLSI Syst* 21(11):2141–2134
49. Hsu CK, Tsai KL, Jheng JF, Ruan SJ, Shen CA (2013) A low power detection routing method for bufferlessNoC. In *Quality Electronic Design (ISQED)*, 2013 14th International Symposium, pp 364–367
50. Lukovic S, Christianos N (2010) Hierarchical multi-agent protection system for NoC based MPSoCs. In *Proceedings of the International Workshop on Security and Dependability for Resource Constrained Embedded Systems, S&D4RCES '10*. ACM, New York, NY, USA, p 6:1–6:7
51. Lukovic S, Christianos N (2010) Enhancing network-on-chip components to support security of processing elements. In: *Proceedings of the 5th Workshop on Embedded Systems Security, WESS '10*. ACM, New York, NY, USA, p 12:1–12:9
52. Ma S, Enright Jerger N, Wang Z (2011) DBAR: an efficient routing algorithm to support multiple concurrent applications in networks-on-chip. In: *Proceedings of the 38th annual international symposium on Computer architecture, ISCA '11*. ACM, New York, NY, USA, pp 413–424
53. Malone M. Talk on OPERA RHBD Multi-core. https://nepp.nasa.gov/mapld_2009/talks/083109_Monday/03_Malone_Michael_mapld09_pres_1.pdf
54. Obermaier R, Hoftberger O (2011) Fault containment in a reconfigurable multi-processor System-on-a-Chip. In: *Industrial Electronics (ISIE)*, 2011 IEEE International Symposium on, pp 1561–1568
55. Munirathinam R, Ponnan S, Chakraborty C. et al. Improved performance on seizure detection in an automated electroencephalogram signal under evolution by extracting entropy feature. *Multimed Tools Appl*;81:13355–13370
56. Otte WR, Dubey A, Pradhan S, Patil P, Gokhale A, Karsai G, Willemsen J (2013) F6com: A component model for resource-constrained and dynamic space-based computing environments. In: *16th IEEE International Symposium on Object/Component/Service-oriented Real-time Distributed Computing*
57. Jaeger T, Sailer R, Sreenivasan Y (2007) Managing the risk of covert information flows in virtual machine systems. In: *ACM Symposium on Access Control Models and Technologies*, France
58. Suresh P, Mariyal C, Rajesh KB, Pillai TV, Jaroszewicz Z (2014) Tightly focusing of spirally polarized Quadratic Bessel Gaussian beam through a dielectric interface. *Optik* 125(3):1264–1266
59. Thirumalai C, Mohan S, Srivastava G (2020) An efficient public key secure scheme for cloud and IoT security. *Comput Commun* 150:634–643
60. Sabelfeld A, Myers AC (2003) Language-based information-flow security. *IEEE J Sel Areas Commun* 21:2003
61. Krohn M, Tromer E (2009) Noninterference for a practical difc-based operating system. In: *Proceedings of the 2009 IEEE Symposium on Security and Privacy*
62. Klein G, Elphinstone K, Heiser G, Andronick J, Cock D, Derrin P, Elkaduwe D, Engelhardt K, Kolanski R, Norrish M, Sewell T, Tuch H, Winwood S (2009) sel4: formal verification of an os kernel. In *SOSP '09: 22nd Symposium on Operating Systems Principles*, pp 207–220, NY, USA
63. Seth B, Dalal S, Jaglan V, Le D-N, Mohan S, Srivastava G (2022) Integrating encryption techniques for secure data storage in the cloud. *Emerging telecommunication technology*, Wiley 2020:1–24. <https://doi.org/10.1016/j.matpr.2021.01.864>
64. Lee JW, Ng MC, Asanovic K (2008) Globally-Synchronized Frames for Guaranteed Quality-of-Service in On-Chip Networks. In *Proceedings of the 35th Annual International Symposium on Computer Architecture*, pp 89–100. Beijing: IEEE
65. Lis M, Shim KS, Cho MH, Ren P, Khan O, Devadas S (2010) DARSIM: A Parallel Cycle-level NoC Simulator. In: *Eeckhout L, Wenisch T (eds) MoBS 2010 - Sixth Annual Workshop on Modeling, Benchmarking and Simulation*, Saint Malo, France
66. Hošek P, Pop T, Bureš T, Hnetyka P, Malohlava M (2010) Comparison of Component Frameworks for Real-Time Embedded Systems. In: *Grunske L, Reussner R, Plasil F (eds) Component-Based Software Engineering*, ser. *Lecture Notes in Computer Science*. Springer Berlin, Heidelberg, pp 21–36. vol. 6092
67. Genßler T, Christoph A, Winter M, Nierstrasch O, Ducasse S, Wuyts R, Arévalo G, Schönhage B, Müller P, Stich C (2002) "Components for Embedded Software: the PECOS Approach," in *Proceedings of the 2002 International Conference on Compilers, Architecture, and Synthesis for Embedded Systems*. New York: ACM, pp 19–26
68. Nierstrasch O, Arévalo G, Ducasse S, Wuyts R, Black A, Müller P, Zeidler C, Genssler T, Van Den Born R (2002) A Component Model for Field Devices, Component Deployment, pp 1–13
69. Wang N, Schmidt DC, Gokhale A, Rodrigues C, Natarajan B, Loyall JP, Schantz RE, Gill CD (2004) QoS-enabled Middleware. In: *Mahmoud Q (ed) Middleware for Communications*. Wiley and Sons, New York, pp 131–162

70. Schmidt DC, Natarajan B, Gokhale A, Wang N, Gill C (2002) TAO: A Pattern-Oriented Object Request Broker for Distributed Real-time and Embedded Systems. *IEEE Distributed Systems Online*;3(2)
71. Benini L, De Micheli G (2002) Networks on chips: a new SoC paradigm. *IEEE Comput* 35–1:70–78
72. Gratz P, Grot B, Keckler S (2008) Regional congestion awareness for load balance in networks-on-chip. In: *Proceedings of 14th International Symposium on High Performance Computer Architecture*. pp 203–214
73. Dally WJ, Seitz CL (1987) Deadlock-free message routing in multiprocessor interconnection networks. *Comp IEEE Trans* 100(5):547–553
74. Duato J (1995) A necessary and sufficient condition for deadlockfree adaptive routing in wormhole networks. *Parallel Distrib Syst, IEEE Trans* 6(10):1055–1067
75. Li M, Zeng Q, Jone W (2006) "DyXY- a proximity congestionaware deadlock-free dynamic routing method for Network on Chip," *Proc. 43th Design Automation Conference (DAC)*, pp. 849–852
76. Xu W, Bhatkar S, Sekar R (2006) Taint-enhanced policy enforcement: a practical approach to defeat a wide range of attacks. In: *15th USENIX Security Symposium*, Vancouver, BC, Canada
77. Light Weight CORBA Component Model Revised Submission, OMG Document realtime/03–05–05 ed., Object Management Group, (2003)
78. Object Management Group, DDS for Lightweight CCM Version 1.0 Beta 2, OMG Document ptc/2009–10–25 ed., Object Management Group (2009)

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)