Proceedings of the 2023 International Conference on Software Engineering and Machine Learning
DOI: 10.54254/2755-2721/8/20230123

An overview on on-chip network routing optimisation

Huyan Tong

College of telecommunication engineering, Xidian University , Xi'an, Shaan Xi , 710126, China

th2010@hw.ac.uk

Abstract. As the number of cores in multi-core systems increases, bus-based systems face significant challenges in terms of scalability, average transmission latency and power consumption. In this context, on-chip networks emerged as a suitable communication architecture for System On Chip (SoC) based entirely on the communication ideas in computer networks and taking into account the characteristics of the system on chip in SoCs. With the introduction of on-chip networks, related research has been developed, such as on-chip network topology, communication quality of service, on-chip network routing algorithms, and on-chip network fault tolerance. He then introduces on-chip network routing algorithms and fault-tolerant routing algorithms from different perspectives, and finally points out the directions of fault-tolerant routing algorithms worthy of research.

Keywords: on-chip networks, network topologies, fault tolerance, routing algorithms.

1. Introduction

To date, the number of transistors in processors has reached up to tens of billions. On the one hand, this increase in transistor count has greatly improved the performance of single-core processors, but on the other hand, it has created many problems and challenges for the design of processor architectures, including increased power consumption, reduced resource utilisation and reduced reliability. In addition, the benefits of improving single-core processor performance by increasing the processor's main frequency, using advanced instruction sets and using large cache arrays are diminishing compared to the dramatic increase in power consumption. Therefore, in order to use such a large number of transistors efficiently while ensuring low power consumption and to further increase processor performance, multi-core processor design has become a key direction to address this issue. Although the industry has made significant progress in the research and practice of multi-core processors (more than 8 cores), there are still many design challenges facing multi-core processors. Highly efficient and low-power on-chip interconnects are one of these challenges. Traditional interconnect structures (e.g. buses or cross-switches) suffer from low scalability, low bandwidth, high latency and high power consumption[1] . Chip networks offer a new way to solve these problems.

Chip Network is a new method of interconnecting systems on chip[2] . It is also a key technical component of multicore on-chip systems. Chip networks provide a new way of on-chip communication between cores, with significantly better performance than traditional bus systems. Network-based on-chip systems can be better adapted to the asynchronous and local global clock synchronisation mechanisms used in complex multi-core SoC designs. Networks in chip architectures

DOI: 10.54254/2755-2721/8/20230123

are mainly based on electronic or optical technologies and are referred to as Network-on-Chip and Optical Network-on-Chip. With advances in semiconductor technology and increased chip integration, existing network theories in chip design have gradually become a reality and show great promise.

2. Related overview

2.1. Basic concepts

An on-chip network is an on-chip communication network that integrates a large number of computing resources into a single chip and connects these resources. The on-chip network consists of two subsystems: computer science and communication. The computational subsystem performs the general task of "computing"; PEs can be CPUs and SoCs in the existing sense, IP cores or memory matrices with various special functions, reconfigurable hardware, etc. The communication subsystem is responsible for connecting the PEs and performing high-speed communication between computing resources. A network consisting of communication nodes and their interconnects is called an on-chip network (OCN), which takes the communication model of a distributed computer system as a reference and replaces the traditional on-chip bus with packet routing and switching techniques for communication tasks [3]. Depending on the on-chip interconnect model, multicore SoCs can be divided into two categories: traditional bus-based interconnects and network-based interconnects. The first is an extension of existing SoCs. Multiple processor cores are integrated into the chip through multi-bus and hierarchical bus technologies to achieve high complexity and performance; the latter is a new concept introduced in recent years where intra-chip communication between multi-processor cores uses packet routing to overcome various bottlenecks caused by bus interconnects; this mode of on-chip communication is known as on-chip networking.

2.2. Advantages of on-chip network routing technology

2.2.1. Contributes to increased communication bandwidth. The bus structure is the communication backbone of existing chip architectures. As circuit sizes increase, the bus structure will become a bottleneck in chip design. While the bus can efficiently connect multiple parts of the communication, the bus address capability cannot scale infinitely with the number of computer drives; while the bus can be shared by multiple users, the bus cannot support multiple pairs of users communicating simultaneously, i.e. the serial access engine leads to a communication bottleneck. In addition, communication on the chip is a major source of power consumption, and the power consumption of the huge clock and bus network will represent the majority of the total power consumption of the chip.

The network topology of the chip network offers good scalability and the chip network provides good parallel communication capabilities, which increases the communication bandwidth by various orders of magnitude. In addition, the chip network converts long interconnections between switches into short interconnections, which is very useful for controlling power consumption. On the other hand, on-chip networks are based on the idea of layering in communication protocols, which allows controlling the total energy consumption from the physical layer to the application layer.

2.2.2. Good for enhancing reuse design. Scalability and reusability of bus architectures are scarce. As a result, when the computing power of the chip evolves, the design must change in line with the processing power requirements (e.g. larger memory widths, higher frequencies, more flexible synchronous or asynchronous designs, etc.). The introduction of each generation of chips is accompanied by varying degrees of design change, which can be quite burdensome for developers. If the communication architecture is designed independently and the most flexible technologies are applied, this will help to shorten the design cycle and reduce development costs. As the level of communication protocol used by the chip network is a separate resource, it provides an architecture that supports an efficient and reusable design approach. Current SoC scales can be "plug and play" as a compute node in a chip-based network node based on a chip-based communication protocol[4] ;

DOI: 10.54254/2755-2721/8/20230123

given the topology of the interconnect, chip integration can be accomplished using an on-chip communication-based design approach. The complete separation between communication technology and computer science (i.e. orthogonal communication and computer design) extends the reusability of reusable computing units to the level of reusability of computing and communication units, thus greatly increasing the level of reusable designs.

3. On-chip network routing technical issues

3.1. Storage structure issues

In existing on-chip multiprocessor systems, memory accounts for 70% of the chip area, which will increase to 90% in the near future. From a power consumption point of view, the power introduced by memory can also reach 90% of the system power consumption, which leads to serious problems in terms of heat dissipation, packaging and chip reliability; on-chip network systems require a large number of storage elements and are organised into a complex storage subsystem that will support parallel data storage, transmission and on-chip network switching [5]. The large number of network storage resources on the chip will occupy multiple routing nodes and, because data is exchanged so frequently between processing units and storage resources, will result in large communication delays if the number of routing nodes in the packet transmission path is excessive. How to effectively reduce the distance between the source and target nodes is crucial to improving the performance of the entire network in a system on chip.

In addition, from a communication bandwidth perspective, as technology advances, the rate of access to computer memory is even higher, meaning that fewer application algorithms achieve near-peak performance based on this architecture. This introduces a number of questions; how many processor cores have enough data to compute? How can the limited on-chip storage space be fully utilised to enable inter-core sharing and avoid off-chip memory accesses? How can the limited memory access bandwidth be fully utilised and attempts be made to allow memory access channels to prioritise access requests to critical path processor cores? Recently, Sandia National Laboratories in the US has proposed stacking memory chips on top of multi-core processor chips to address the lack of bandwidth growth, which may be a viable solution. In summary, on-chip storage architecture has become one of the major factors affecting the performance of on-chip networks.

3.2. Software parallelisation issues

Future-based high-performance multicore processing chips may encounter many applications where traditional methods of automatic parallelisation of serial programs are difficult to achieve. The actual performance of parallel computing will be very low if the parallel processing power of the network on the chip is not effectively utilised. Therefore, it becomes an urgent problem to make full use of the many network processing units on the chip through effective methods and models and to significantly reduce the difficulty of application development.

Similar to the problems encountered in parallel computer development, the main problem faced by networks in parallel chip processing architectures is how to efficiently extract the different layers and granularities of parallelism contained in an application and map them to a multi-core parallel hardware architecture. Many aspects are involved in solving this problem, including programming models, programming languages, compilation systems and hardware support.

In general, there are three approaches to developing parallel programs: firstly, the automatic parallelisation of serial programs (which is not yet complete), the most realistic goal of which should be automatic parallelism in human-computer interaction, and secondly, the design of a new parallel programming language. The disadvantage of this approach is that it requires a complete rewrite of the original program, which is also costly and risky for the user, and efficiency cannot be guaranteed. However, with the advent of multi-core, a new and accessible programming language is necessary if parallel computing environments are to be popular with the general public. The emerging parallel programming languages currently being investigated around the world, such as IBM X10, UPC

DOI: 10.54254/2755-2721/8/20230123

(Unified Parallel C, C extensions) and Titanin (Java extensions), i, e. Add a library or some new guide instructions to help with information transfer and parallelism[6] . This is exactly the approach used by MPI and OpenMP and is currently a relatively acceptable high performance approach, but its program development is very inefficient and relatively difficult .[7]

3.3. Power management issues

While on-chip meshes can help improve the energy efficiency of a chip, it cannot be ignored that the issue of power consumption still hangs in the balance due to the significant increase in integration size on multi-core system-on-chips. How to improve energy efficiency and program and manage the large amount of computational resources to minimise power consumption in on-chip network design remains one of the key issues facing on-chip network design.

Architecturally, the network-on-a-chip consists of three main components: the core processor, the inter-core interconnect and the on-chip storage. Research on low power consumption in chip networks can be focused on four areas: power evaluation, processor core power optimization , chip network power optimisation and storage in chip power optimization [8] .

Power consumption is an important trigger for the emergence of multi-core technologies, including chip networks, and an important constraint for multi-processor chip designs. There are effective ways to reduce power consumption for different design modules and chip network design levels, and these can be limited and influenced by each other. Therefore, you need to have extensive knowledge of all aspects of chip network architecture to circuit technology to make the right choice of multicore architecture early in your project. In general, the higher the level of abstraction of a project, the more efficient it is to reduce energy consumption.

4. On-chip network routing optimization

4.1. Fault-tolerant routing algorithms

Fault tolerant routing algorithms are a routing algorithm for normal communication on a faulty on-chip network[9] . For a potentially faulty on-chip network, this can be solved by sending the same packet repeatedly to a different path, or by sending the packet to a bypass on the faulty node. For paths that send individual packets, they can be divided into different types of routes depending on the granularity of the fault: routing error blocks. The centralised error node and surrounding nodes are considered to be the entire error block, with packets spinning single path failures around the error block during transmission. Error nodes are treated as individual nodes, regardless of the connectivity of the error node[10] . Based on the dimensional sequential routing algorithm, when a single error node is found, the form of deviation in its natural boundaries is used. To avoid deadlocks, partial deviation patterns need to be changed[11] . The granular disruption process it divides network faults on the chip into node faults and connection faults, which makes implementing the routing algorithm more complex but reduces the waste of resources on the chip.

4.2. Source routing algorithms

Source routing algorithms are mainly used for macro networks. Stores network status information, including congestion and fault conditions, across network nodes. When a network problem occurs, the node sends packets to nearby nodes, which return link packets upon receiving them to determine if the network is faulty. If a network failure occurs, packets are sent to all nodes in the on-chip network and the entire network information table is updated. If a routing decision is required, the best path to the destination node is found based on the node information (faults, delays, congestion, etc.) stored at the source node[12] . The source routing algorithm is the source node that completes all routing decisions. The source node adds the routing path to the packet header. The network node forwards packets based on the routing information in the header and is no longer involved in the routing decisions. The advantage of the source routing algorithm is that it has global connectivity information when searching for routing decisions, so the routing policy can integrate various information to find the best route.

DOI: 10.54254/2755-2721/8/20230123

However, due to the complexity of implementing the algorithm and storing network information, it is not suitable for network chips with simple node structures.

5. Conclusion

From a communication bandwidth perspective, as technology advances, access rates to computer memory are even higher, meaning that fewer application algorithms achieve near-peak performance based on this architecture. This introduces many questions as to how many processor cores have enough data to compute. How to make full use of the limited on-chip storage space for inter-core sharing and avoid off-chip memory accesses. How to make full use of the limited memory access bandwidth and try to make memory access channels prioritise access requests to critical path processor cores. Recently, stacking memory chips on multi-core processor chips has been proposed as a possible solution to the problem of insufficient bandwidth growth. In summary, the on-chip storage architecture has become one of the major factors affecting the performance of on-chip networks.

References

[1] Zhang DK, Huang C & Song GZ. (2016). A review of research on three-dimensional on-chip networks. Journal of Software (01), 155-187.

[2] Wang J, Li YB & Peng QW. (2011). Optimal design of routing nodes for on-chip networks. Computer Applications (03), 617-620.

[3] Sun, F. & Liu, Y. J.. (2017). Analysis and optimal design of an on-chip network routing algorithm. Journal of Guangdong University of Technology (05), 60-64.

[4] Jin To. (2016). A Review of Research on Wireless On-Chip Networks and Optimization of Broadcast and Sink Communications. Master thesis of Nanjing University, 8: 78.

[5] Yang Hongbin, Zhao Yaqian, Dong Gang (2020) An on-chip network routing optimization method, apparatus, device and readable storage medium:https://www.xjishu.com/zhuanli/62/201911082354.html

[6] Wang, Chunlai. (2016). Network evolution model and performance optimization of multi-path shortest route NoC. Master thesis of Hefei University of Technology, 5: 74.

[7] Jiang, Haiyan. (2022). Analysis and optimization of storage device capacity and router performance in on-chip routers. Electronic Technology and Software Engineering (10), 22-25.

[8] He Q. (2011). Research on Optimized Design of On-Chip Networks Based on Voltage Islands. Master thesis of University of Electronic Science and Technology, 7: 87.

[9] Chen, Jiahao. (2021). On-chip network routing optimization for multicore cache coherency. Master thesis of University of Electronic Science and Technology, 1: 74.

[10] Jingcheng Shao. (2014). Performance Optimization of On-Chip Networks with Accelerated Networks. Master thesis of Zhejiang University, 6: 108.

[11] Wang, Kunpeng. (2013). Research on Application-Oriented Routing Algorithms for On-Chip Networks. Master thesis of Xi'an University of Electronic Science and Technology, 4: 60.

[12] Cao, Meng. (2009). A new routing algorithm for on-chip hybrid mesh networks. Learning (4), 15-15.