A Hybrid Multicast Routing Algorithm For Networks-On-Chip For Improving Connectivity

B. Thenral[1], D. Santhosh kumari[2], P. Iswarya[3], K. Arthi [4]
Assistant Professor, Ganadipathy Tulsi's Jain Engineering College

Abstract - In order to achieve better performance, Multicast communication has been commonly used in Multiprocessor systems. In NoC-based designs, the communication systems, often referred to as communication networks, are composed of modular components such that high scalability can be achieved. Two main challenges exist for multicast communication. One is how to deliver multi-destination packets efficiently and the other is how to resolve the deadlock problem that may occur when two or more packets form a circular wait and hold loop. In this paper, a multicast routing algorithm that is much more efficient than the currently available multicast routing algorithms and can completely solve the multicast deadlock problem is proposed. The main idea of the proposed hybrid multicast routing is to combine the advantages of the tree-based and path-based methods such that the routing can vary adaptively according to the traffic of the network. The path-based multicast routing only determines the forwarding direction of the first head flit; the other flits of the packet just follow the first head flit. In the proposed hybrid multicast routing, the mainstream of the multicast packet delivery is a path, but there can be some branches from the mainstream as long as the branch will not cause deadlocks. Very high routing efficiency is achieved by the proposed algorithm due to an adaptive routing strategy according to the traffic load and also connectivity holes are avoided by implementing the rainbow mechanism in this routing algorithm. Simulations in the network simulator 2 prove the efficiency of the proposed scheme.

Keywords – multicast, deadlock, hybrid multicast routing.

1. INTRODUCTION

The main idea of the proposed hybrid multicast routing is to combine the advantages of the tree-based and path-based methods such that the routing can vary adaptively according to the traffic of the network. As mentioned in Section 2, the path-based multicast routing only determines the forwarding direction of the first head flit; the other flits of the packet just follow the first head flit. In the proposed hybrid multicast routing, the mainstream of the multicast packet delivery is a path, but there can be some branches from the mainstream as long as the branch will not cause deadlocks. In this way we can 1) shorten the length of the transmission paths, 2) increase the usage of y-direction channels such that the congestion of x-direction channels can be alleviated, and 3) avoid multicast deadlocks. NS 2.28 Simulation tool along with Constant bit rate (CBR) traffic model is used to evaluate the performance of the proposed scheme. The Object oriented Tool Command Language is used for programming and configuring the network scenario.

Network on a Chip:

Network on chip or network on a chip (NoC or NOC) is a communication subsystem on an integrated circuit (commonly called a "chip"), typically between IP cores in a system on a chip (SoC). NoCs can span synchronous and asynchronous clock domains or use unclocked asynchronous logic. NoC technology applies networking theory and methods to on-chip communication and brings notable improvements over conventional bus and crossbar interconnections. NoC improves the scalability of SoCs, and the power efficiency of complex SoCs compared to other designs.

Trend of NoC's:

Network on chip is an emerging paradigm for communications within large VLSI systems implemented on a single silicon chip. Sgroi et al. call "the layered-stack approach to the design of the on-chip inter-core communications the network-on-chip (NOC) methodology." In a NoC system, modules such as processor cores, memories and specialized IP blocks exchange data using a network as a "public transportation" sub-system for the information traffic. A NoC is constructed from multiple point-to-point data links interconnected by switches (a.k.a. routers), such that messages can be relayed from any source module to any destination module over several links, by making routing decisions at the switches. A NoC is similar to a modern telecommunications network, using digital bit-packet switching over multiplexed links. Although packet-switching is sometimes claimed as necessity for a NoC, there are several NoC proposals utilizing circuit-switching techniques. This definition based on routers is usually interpreted so that a single shared bus, a single crossbar switch or a point-to-point network are not NoCs but practically all other topologies are. This is somewhat confusing since all above mentioned are networks (they enable communication

between two or more devices) but they are not considered as network-on-chip approaches.

The wires in the links of the NoC are shared by many signals. A high level of parallelism is achieved, because all links in the NoC can operate simultaneously on different data packets. Therefore, as the complexity of integrated systems keeps growing, a NoC provides enhanced performance (such as throughput) and scalability in comparison with previous communication architectures (e.g., dedicated point-to-point signal wires, shared buses, or segmented buses with bridges). Of course, the algorithms must be designed in such a way that they offer large parallelism and can hence utilize the potential of NoC.

Benefits of adopting NoCs:

Traditionally, ICs have been designed with dedicated point-to-point connections, with one wire dedicated to each signal. For large designs, in particular, this has several limitations from a physical design viewpoint. The wires occupy much of the area of the chip, and in nanometer CMOS technology, interconnects dominate both performance and dynamic power dissipation, as signal propagation in wires across the chip requires multiple clock cycles.

NoC links can reduce the complexity of designing wires for predictable speed, power, noise, reliability, etc., thanks to their regular, well controlled structure. From a system design viewpoint, with the advent of multi-core processor systems, a network is a natural architectural choice. A NoC can provide separation between computation and communication, support modularity and IP reuse via standard interfaces, handle synchronization issues, serve as a platform for system test, and, hence, increase engineering productivity.

Although NoCs can borrow concepts and techniques from the well-established domain of computer networking, it is impractical to blindly reuse features of "classical" computer networks and symmetric multiprocessors. In particular, NoC switches should be small, energy-efficient, and fast. Neglecting these aspects along with proper, quantitative comparison was typical for early NoC research but nowadays they are considered in more detail. The routing algorithms should be implemented by simple logic, and the number of data buffers should be minimal. Network topology and properties may be application-specific.

Some researchers think that NoCs need to support quality of service (QoS), namely achieve the various requirements in terms of throughput, end-to-end delays and deadlines. Real-time computation, including audio and video playback, is one reason for providing QoS support. However, current system implementations like VxWorks, RT Linux or QNX are able to achieve sub-millisecond real-time computing without special hardware. This may indicate that for many real-time applications the service quality of existing on-chip interconnect infrastructure is sufficient, and dedicated hardware logic would be necessary to achieve microsecond precision, a degree that is rarely needed in practice for end users (sound or video jitter need only tenth of milliseconds latency guarantee). Another motivation for NoC-level quality-of-service is to support multiple concurrent users sharing resources of a single chip multiprocessor in a public cloud computing infrastructure. In such instances, hardware QOS logic enables the service provider to make contractual guarantees on the level of service that a user receives, a feature that may be deemed desirable by some corporate or government clients.

To date, several prototype NoCs have been designed and analyzed in both industry and academia but only few have been implemented on silicon. However, many challenging research problems remain to be solved at all levels, from the physical link level through the network level, and all the way up to the system architecture and application software. The first dedicated research symposium on networks on chip was held at Princeton University, in May 2007. The second IEEE International Symposium on Networks-on-Chip was held in April 2008 at Newcastle University.

Research has been done on integrated optical waveguides and devices comprising an optical network on a chip (ONoC).

2. LITERATURE SURVEY

The scaling of microchip technologies has enabled large scale systems-on-chip (SoC).[1] Network on-chip (NoC) research addresses global communication in SoC, involving (i) a move from computation-centric to communication-centric design and (ii) the implementation of scalable communication structures. This survey presents a perspective on existing NoC research. We define the following abstractions: system, network adapter, network, and link to explain and structure the fundamental concepts. First, research relating to the actual network design is reviewed. Then system level design and modeling are discussed. We also evaluate performance analysis techniques. The research shows that NoC constitutes a unification of current trends of

intrachip communication rather than an explicit new alternative.Current state-of-the-art on-chip networks provide efficiency, high throughput, and low latency for one-to-one (unicast) traffic.[2] The presence of one-to-many (multicast) or one-to-all (broadcast) traffic can significantly degrade the performance of these designs, since they rely on multiple unicasts to provide one-to-many communication. This results in a burst of packets from a single source and is a very inefficient way of performing multicast and broadcast communication. This inefficiency is compounded by the proliferation of architectures and coherence protocols that require multicast and broadcast communication. In this paper, we characterize a wide array of on-chip communication scenarios that benefit from hardware multicast support. We propose Virtual Circuit Tree Multicasting (VCTM) and present a detailed multicast router design that improves network performance by up to 90% while reducing network activity (hence power) by up to 53%. Our VCTM router is flexible enough to improve interconnect performance for a broad spectrum of multicasting scenarios, and achieves these benefits with straightforward and inexpensive extensions to a state-of-the-art packet-switched router.

A novel hardware supporting for multicast on mesh Networks-on-Chip (NoC) is proposed.[3] It supports multicast routing on any shape of tree-based path. Two power-efficient tree-based multicast routing algorithms, Optimized tree(OPT) and Left-XY-Right-Optimized tree (LXYROPT) are also proposed. XY tree-based(XYT) algorithm and multiple unicast copies (MUC) are also implemented on the router as baselines. Along with the increase of the destination size, compared with MUC, OPT and LXYROPT achieve a remarkable improvement both in latency and throughput while the average power consumption is reduced by 50% and 45%. Compared with XYT, OPT is 10% higher in latency but gains 17% saving in power consumption. LXYROPT is 3% lower in latency and 8% lower in power consumption. In some cases, OPT and LXYROPT give power saving up to 70% less than the XYT.

We show that deadlocks due to dependencies on consumption channels are a fundamental problem in multicast wormhole routing. This issue of deadlocks has not been addressed in many previously proposed multicast algorithms We also show that deadlocks on consumption channels can be avoided by using multiple classes of consumption channels and restricting the use of consumption channels by multicast messages .In addition we present a new multicast routing algorithm column path which uses the well known cube algorithm for multicast routing. Therefore this algorithm could be implemented in the existing multi computers with minimal hardware support. [4]We present a simulation study of the performance of Hamilton path based multicast algorithms with the proposed column path algorithm. Our simulations indicate that the simplistic scheme of sending one copy of a multicast message to each of its destinations exhibits good performance and that the new column path algorithm others higher throughput compared to the Hamilton path based algorithms.

Nowadays, in MPSoCs and NoCs, multicast protocol is significantly used for many parallel applications such as cache coherency in distributed shared-memory architectures, clock synchronization, replication, or barrier synchronization. Among several multicast schemes proposed in on chip interconnection networks, path-based multicast scheme has been proven to be more efficient than the tree-based, and unicast-based. In this paper a low distance path-based multicast scheme is proposed[5]. The proposed method takes advantage of the network partitioning, and utilizing of an efficient destination ordering algorithm. The results in performance, and power consumption show that the proposed method outstands the previous on chip path-based multicasting algorithms.

3. DESIGN & MODULES OF THE HYBRID MULTICAST ROUTING ALGORITHM

Modules

The entire implementation and research work in this project can be split into the following modules in order to obtain the required results:

1. System configuration
2. Hybrid multicast routing algorithm
3. Design Implementation
4. Performance Evaluation

System configuration

The needed number of nodes is generated by using the node command in NS2. The nodes are disseminating in a wireless environment. The random motion is set as false. So, the nodes are moving fixed as in a NoC. Each node is considered as an autonomous node. The nodes are configured as to process in Mesh Network environment. The node configuration is done by using node-config command. We have to specify the Channel used by the node, Radio propagation model, Link layer type, Physical layer type,

Type of interface queue and the protocol used to route the packets dynamically.

Hybrid multicast routing algorithm

All head flits of a packet are transmitted according to the following label routing rule: A flit will always be routed to a node with a smaller label if the label of its destination node is smaller than the label of the source node; otherwise, it is routed to a node with a larger label. In the proposed hybrid routing algorithm, the first head flit of a packet is called the leading head flit, and the other head flits are called the following head flits. A leading head flit is transmitted along a path just like those path-based routing algorithms, and a following head flit is determined to follow the direction of the leading head flit or branch to a direction different from the leading head flit. If a following head flit branches, the routing is like the tree-based methods. The key of our algorithm is to examine two conditions. One is whether the input channel of the downstream router, i.e., the router that the packet is to be sent to, has enough unused buffer space to accommodate the head flit plus the entire data and tail flits of the packet. The other is whether the head flit will immediately reach its destination after branching. It can be shown that in both cases, the branch will not lead to a deadlock. This algorithm determines the routing direction for each head flit input to the algorithm. It can be divided into two parts: for a leading head flit and for a following head flit.

Design Implementation

Once the system configuration is done and the hybrid multicast routing algorithm is designed, the design is implemented in Object Oriented Tool Command Language (OTCL). Object TCL (OTCL) is an object oriented extension of TCL. To write a network simulation program in NS2, the programmer simply write an OTcl program that creates network objects and one (special) simulator object. In this project, Network Simulator ns-2.28 is used with OTCL to simulate the design and obtain the animated output from the Network Animator (NAM).

Performance evaluation

During simulation time the events are traced by using the trace files. The performance of the network is evaluated by executing the trace files. The events are recorded into trace files while executing record procedure. In this procedure, we trace the events like latency, packet received, Packets lost, etc. These trace values are write into the trace files. This procedure is recursively called for every 0.05 ms. so, trace values recorded for every 0.05 ms.

4. CONCLUSION

A multicast routing algorithm that is much more efficient than the currently available multicast routing algorithms and can completely solve the multicast deadlock problem was proposed. The advantages of the tree-based and path-based methods such that the routing can vary adaptively according to the traffic of the network are combined to form the proposed scheme. Very high routing efficiency is achieved by the proposed algorithm due to an adaptive routing strategy according to the traffic load. Simulations in the network simulator 2 prove the efficiency of the proposed scheme. Performance evaluation of the proposed scheme is demonstrated by the ns 2.28 simulations by programming in OTCL. The Xgraph plots are obtained and stand as evidence of the proposed hybrid multicast routing algorithm.

5. FUTURE SCOPES

The basic idea for avoiding connectivity holes is that of allowing the nodes to forward packets away from the sink when a relay offering advancement toward the sink cannot be found. In order to remember whether to seek for relays in the direction of the sink or in the opposite direction each node is labeled by a color chosen among an ordered list of colors, and searches for relays among nodes with its own color or the color immediately before in the list. Rainbow consists in determining the color of each node so that a viable route to the sink is always found.

REFERENCES

[1] T. Bjerregaard and S. Mahadevan, "A survey of research and practices of Network-on-chip," ACM Comput. Surv., vol. 38, no. 1, pp. 1-51, 2006.

[2] N.E. Jerger, L.S. Peh, and M. Lipasti, “Virtual Circuit Tree Multicasting: A Case for On-Chip Hardware Multicast Support,” Proc. Int’l Symp. Computer Architecture (ISCA ’08), pp. 229-240, June 2008.

[3] W. Hu, Z. Lu, A. Jantsch, and H. Liu, “Power-efficient tree-based multicast support for Networks-on-Chip,” in Proc. Asia and South Pacific Des. Autom. Conf., Jan. 2011, pp. 363-368.

[4] X. Lin and L.M. Ni, “Multicast communication in multicomputer networks,’’ IEEE Trans. Parallel and Distrib. Syst., vol. 4, no. 10, pp. 1,105-1,117, Oct. 1993.

[5] An Efficent Dynamic Multicast Routing Protocol for Distributing Traffic in NOCs Masoumeh Ebrahimi, Masoud Daneshtalab, Mohammad Hossein Neishaburi, Siamak Mohammadi, Ali Afzali-Kusha, Juha Plosila, Hannu Tenhunen.

AUTHOR'S PROFILE

B.THENRAL has received her Bachelor of Engineering degree in Electronics & Communication Engineering from GGR College of Engineering, in the year 2005, received her M.Tech from MGR University, At present he is pursuing PhD with the specialization of mobile adhoc networks in St peter's University. She has been working has a assistant professor in Ganadipathy Tulsi's Jain Engineering College, Vellore for the past 10 years. Her area of interest RF & Microwave engineering, Microprocessors & Microcontrollers, Electronic devices and mobile adhoc networks.