Wormhole Routing In Network on Chip

Elizabeth isaac*1 and Dr. M. Rajasekhara Babu #2

* Research Scholar, V IT University,
Assistant Professor, Department of Computer Science and Engineering,
Rajagiri school of engineering and technology, Kochi, India
 # School o f Computer Science and Engineering,
VIT University, Vellore, Tamil Nadu, India
1 elizabeth.issaac@gmail.com, 2 mrajasekharababu@vit.ac.in

Abstract

Network on Chip is a scalable solution for the communicational problems on System-on-Chip (SoC). Network on Chip was proposed to solve the on-chip communication problem by separating the concerns of communication from computation. Network on Chip is used to develop large and complex systems of same processors on a single chip. There are hundreds of routing algorithms that are developed by the researches. In this paper we are dealing with the switching methodology which specifies how the message can be routed.

Introduction

Ever growing need of the computer industry is to improve the performance in terms of space, reliability, speed of execution and cost. Over the years with the evolution of the different generation of the computer we have moved from the massive ENIAC [1] to modern PC standards. With the growth of VLSI technology, there was an exponential decrease in the size of the processing units. This leads the way for integration of heterogeneous IP cores on a single chip leading to a new era of integration circuits known as System-on-Chip (SoC). SoC is an era of heterogeneous cores in single chip. Integration of more components to the on-chip bus declines the scalability of SoCs. Inter connection between the processing units affects the performance.

Traditional bus-based communication schemes, which lack scalability and predictability, are not capable to cop up with the increasing requirements of future SoCs in terms of performance, power, timing closure, scalability, and so on. To meet the design productivity and signal integrity challenges of next-generation system designs, a structured and scalable interconnection architecture, network on chip (NoC), has been proposed recently to mitigate the complex on-chip communication problem

[2]. The use of multicore chips with hundreds and thousands cores and parallel programming model are the current trends in the computer industry. *Network on Chip* (NoC) is a new paradigm for System *on Chip* (SoC) design.

NOC is a grid of homogeneous nodes. Instead of busses and dedicated point-to-point link the routing nodes are connected by means of communication links similar to the internet. With the increasing number of cores in the chip we can improve the performance but we should think about the routing of the data and the reliability of this network. The intel is dreaming for a tera-scale world with hundreds of cores for massive parallelism for the world's growing mountain of data [3].

In this paper section 2 deals with the NoC architecture, section 3 shows some network topologies, section 4 addresses the switching methodologies, next section deals with the routing algorithms and section 6 draws the network flow control and section 7 deals conclusion.

Network on Chip Architecture

NoC architecture provides the communication infrastructure for the resources [4]. The component that facilitates the communication process in an NOC is the network adapter, link and the router. Network Adapter or Network Interface acts as the interface between the core and the network. It acts as the separation between the routing logic and the communication. Communication link is composed of a set of wires to give the raw band width. Link can be a physical link and one or more logical link channels. Router, the heart of the NoC, contains the buffers and routing logic for routing of the message. Data communications between segments of chip are packetized and transferred through the network.

The terminal node can be a memory, processor or any hardware component. Figure 1 shows the simple mesh interconnection topology [2]. Intellectual property (IP) core module communicates with each other by sending messages. IP module can be computation or storage or their combination.

Each core has a unique address. Unlike the TCP/IP that uses seven, layers the NoCs defines four protocol layers- physical layer, data link layer, network layer and transport layer [13].

NoC is characterized by following important features-topology, switching mechanism, flow control, routers architecture and the routing algorithms. Topology defines how the nodes are interconnected in the network. Routing algorithms involves the logic in selecting a path between the source and destination using a specific switching technique in a particular topology.

Noc Topology

The arrangement of different cores forms the NoC topology. Network topology can be also defined as the static arrangements of the nodes along with the links. Figure 2 shows mesh, torus and tree topology.
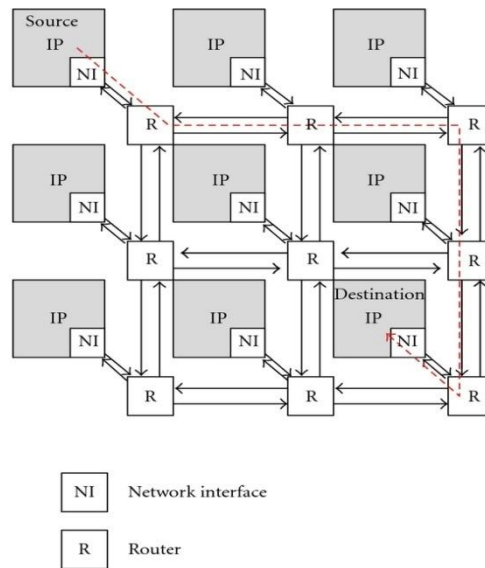
Figure 1: Typical NoC architecture

Mesh is the simplest and most common type of topology. Every switch is connected to four other neighboring switches except the edge switches. The main disadvantage of the 2d mesh is that the number of hops will increase as the size of the system increases. Torus is same as the mesh but the heads of the rows and columns are linked to the tails of the rows and columns. Each node is connected to four other neighbouring switches. In a tree topology each node has replicated ancestors which mean that there are many alternative routes between nodes.

Switching Methedologies

Switching technique determines how and when the messages are routed through the nodes in NoC. Different switching techniques include circuit switching, message switching packet switching and wormhole switching.

In circuit switching a dedicated path exists between the source and the destination. The path once established cannot be modified according to the network congestions. A physical path from source to destination is reserved prior to the data transmission. Advantage is that the network bandwidth is reserved during the entire duration of the data transmission.

However the degree of fault tolerance is less and can cause unnecessary delays.

In message switching there is no dedicated path between the two nodes. When the source node sends the data the destination node address is also sent along with the message. When a message along with the address reaches a router the whole data are stored in the router buffer, the router will not start forwarding the data until whole of the message reaches the router.

(a) Mesh   (b) Torus   (c) Binary Tree
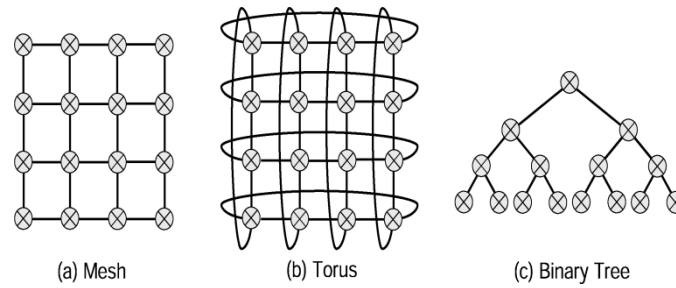
Figure 2: Regular topologies

This is called store and forward technique. Although this technique has greater channel efficiency it has high latency as it needs the time to store the complete message. It needs to find the next node from the address and the buffer size also should be as large as the largest message.

In order to reduce the time to store the packet at each node a virtual cut-through method was introduced in 1979[5]. When the message arrives at any intermediate node, based on the decision of the routing algorithm the output channel is selected. If the output channel is free then in contrast to the message switching the message is send out to the adjacent node before it is received completely at the current node. If the message is blocked due to the channel busy the technique is same as message switching.

In packet switching data is divided into fixed-length block called packet. Each packet is appended with a source and the destination address, so even if the packets have same source and destination address, they need not necessarily travel through the same path. The entire packet needs to be stored before start of transmitting the data to the next node.

In the worm hole switching the packet is again divided into some units called flit-flow control unit. Worm hole switching is the correct choice for NOCs. The flit type field indicates four different options - head, tail, body and single-packet flit. Usually multiple flits constitute a packet and if the packet is having only single flit it is called HDT flit. Header flits only contains the routing information. Header flits establishes the path and the body and the tail flits follow the established path in a pipelined fashion. As a result each incoming data-flit of a message packet is simply forwarded along the same output channel as the preceding data flits. Chances of channel busy for flits cannot be eliminated and in such situations all the successive flits have to wait in their current location. As a result no packet reordering is required at destinations.

Figure 3 shows the Wormhole switching. The message is first divided into packets called packetization and the packets in turn are divided into flits during the process of flitisation. Header flit contains the virtual channel identifier, the flit type, the source address, destination address, sequencing bit, priority and checksum.
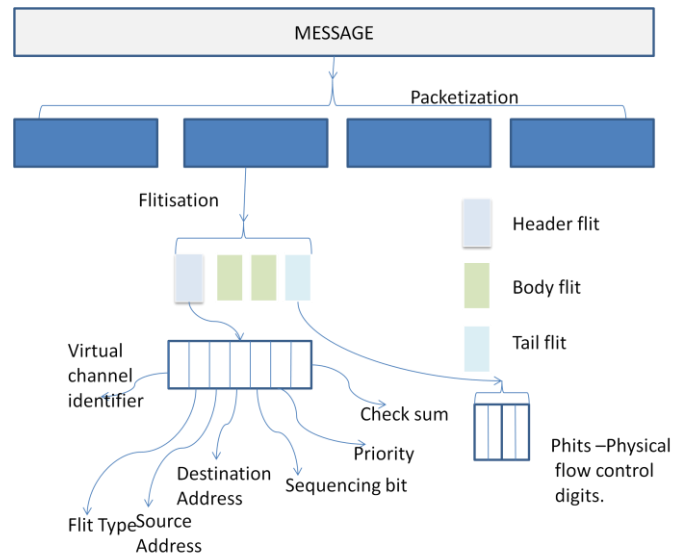
Figure 3: Wormhole switching

Mad Postman Switching is an improvement over worm hole switching. When the header flit reaches a router, the message is routed in the same direction as the incoming flits with the assumption that the message will be continuing in the same direction. The header bits are forwarded to the output port in the same direction. When the last bit in the header bit reaches the current router the payload of the header flit is decoded to get the required direction. If the direction is incorrect, the rest of the message starting with the second flit of the header is transmitted along the required direction. Checking is performed after the routing processes hence the name Mad Postman Switching. A dead flit is generated when the flits changes its direction.

Routing Algorithms

The routing algorithms decide the path of a packet to reach the destination. The routing algorithm must be free from live lock, deadlock, starvation and adaptiveness. In Live lock the packet can advance but will be moving around the destination but never reaches the destination. Deadlock packet cannot advance to the destination, as the packet is waiting for the release of the resource which is held by packet which in turn needs a release. The packet enters in a cycle of resource dependencies. Setting priorities to the packet can often leads to starvation. This happens when the packets with lower priorities never reaches the destination as the higher priorities reserve the resources all the time. Adaptiveness increases the chances that packets may avoid hot spots or faulty components and reduces the chances that packets are continuously blocked [6].

Routing algorithms can be deterministic and adaptive. Deterministic routing algorithms always have the same path between the source and the destination even if multiple paths exist. Deterministic routing requires less computation time to find the next node than the adaptive routing algorithms. Adaptive routing algorithm provides

multiple routing options to forward the packet. Thus, with adaptive routing, congested areas in the network can be avoided. Usually the adaptive algorithms gave better throughput than the deterministic algorithm.

Fat-trees offer a rich connectivity among nodes, making possible to obtain paths between all source-destination pairs that do not share any link. The results of a study on fat-tree based on Deterministic and Adaptive Routing show that deterministic routing can achieve a similar, and in some scenarios higher, level of performance than adaptive routing, while providing in-order packet delivery [7].

In XY routing, the most popular dimension order routing, routes the packet first in the X − direction to correct column and then in Y-direction to reach the destination. YX routing is also a dimension order routing. XY and YX routing are suited for mesh and torus topology. But the main disadvantage is that the algorithm causes biggest load in the middle of the network. The traffic load is not equalized over the whole network. C.J. Glass and L.M. Ni have proposed some of the partially adaptive routing algorithms for the 2D meshes. The model involves analysing the possible direction of turns of a packet and just restricting some turn. The suggested algorithms are West-first, north-last and Negative-first called turn models. West-first routing, routes a packet to west first, if necessary, then to south, east and north. Any turn to other direction is possible if at least the destination column is reached. North-last routing, if necessary, adaptively turns to east, south, west and then at last to north. North turn is possible only if the destination column is reached. Negative first routing prohibits a turn from positive to negative direction. Turn model works for variety of network architecture like mesh-connected, k-ary n-cube, hexagonal, octagonal, and cube connected cycle networks but the degree of the adaptiveness of the turn model is uneven.

The adaptive odd-even turn model is based on restricting the locations at which certain turns can be taken so that a circular wait can never occur [9]. More precisely, the odd-even turn model is governed by the following two rules:

Rule 1. Any packet is not allowed to take an East to North turn at any nodes located in an even column, and it is not allowed to take a North to West turn at any nodes located in an odd column.

Rule 2. Any packet is not allowed to take an East to South turn at any nodes located in an even column, and it is not allowed to take an South to West turn at any nodes located in an odd column.

Odd-even routing works well on non-uniform traffic and it provides better adaptiveness but cannot be used for the fault tolerant routing. Other adaptive routing algorithms include Q-Routing [10] which works on network traffic statistics. Congestion aware routing based on the free slots in the input buffers beyond the adjacent routers and the cross bar demand was described in [11].

To achieve network layer fault tolerance we should make use of fault-tolerant algorithm. There are a number of fault tolerance algorithms available for the NoC architectures. The fault tolerance algorithms look into faulty nodes or routers and the link failures. These failures may be due to the heterogeneous cores, fabrication faults or cross talk.

An efficient routing algorithm is important for large on-chip networks (NoC) to provide the required fault tolerance and reliability along with the required performance specific to the application.
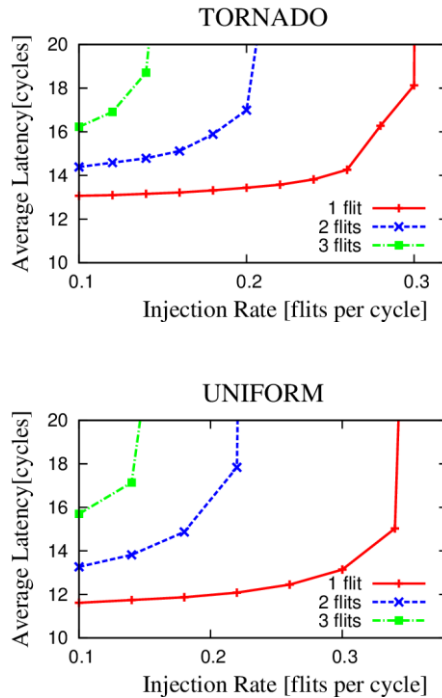
Figure 4: Comparative analysis of average flit latency with injection rate for various traffic patterns mesh network.

Effect on Average Flit Latency

Latency of a flit is defined as the time required to travel the network from the source to destination. Figure 4 shows the plot of injection rate vs packet latency for various traffic patterns in 4x4 mesh network. Average flit latency increases with increase in the injection rate. Average flit latency also increases when the packet size is increased. The graph plot the average flit latency for single flit per packet, two flits packet and three flits per packet.

Network Flow Control-Buffering

Router at each tile consists of five input/output controllers (one for each direction and one for the tile). Each input controller has an input buffer. When the header flits arrives at each of the tiles the input controller decodes the destination address on the header flit to select the one out of four output port.

In Noc the most of the power used is for the buffers and the link. About 25-30% of the power is used by the link and 40-50% of the power is used by the buffer [12]. So bufferless routing can be a good choice for reducing the power consumption.

Conclusions

Network on Chip is a platform for single chip systems which scales well to an arbitrary number of processor like resources. It is concluded that wormhole routing is a good switching technique for buffered and buffer-less NoCs. With wormhole routing flit level routing is possible. Flits of single packet can be a good choice for buffer-less routing as the reordering of the packet at the destination is cost consuming. It is also concluded that packet size affects the performance of the network.

References

[1] http://en.wikipedia.org/wiki/ENIAC

[2] T. Wen-Chung et al., "Networks on Chips: Structure and Design Methodologies",Hindawi Publishing Corporation Journal of Electrical and Computer Engineering , 2012.

[3] From a Few Cores to Many: "A Tera-scale Computing", intel.

[4] Shashi Kumar, Axel jantsch,, "A Network on Chip Architecture and Design Methodology", Proceeding of the IEEE.

[5] P.Kermani and L.Kelinrock, "Virtual Cut-through:A computer communication Switch technique," Computer Network, vol.3, pp. 267-286, 1979.

[6] C. J. Glass and L. M. Ni, "The Turn Model for Adaptive Routing," J. Assoc. for Computing Machinery, vol. 41, pp. 874-902, 1994.

[7] C. G´omez et al., "Deterministic versus Adaptive Routing in Fat-Trees" IEEE 2007.

[8] C.J. Glass and L.M. Ni, ªThe Turn Model for Adaptive Routing,º Proc. 19th Ann. Int'l Symp. Computer Architecture, pp. 278-287, May 1992.

[9] G.M. Chiu "The Odd-Even Turn Model for Adaptive Routing", IEEE Transactions on Parallel and Distributed Systems, VOL. 11, NO. 7, July 2000

[10] M.Majer, et al., "Packet Routing in Dynamically Changing Networks on Chip". Proceedings, 19th IEEE International Parallel and Distributed Processing Symposium, 4–8 April 2005, page: 154b.

[11] Dana .A, Salehi N "Congestion aware routing algorithm for mesh network on chip platform" in Indian Journal of Science and Technology, Vol.5, pp. 2822- 2830, 2012.

[12] Srinivasan, K. and K. S. Chatha, "A low complexit heuristic for design of custom network-on-chip architectures", in: Proceedings of DATE, 2006

[13] Shashi Kumar et al., "Network-on-chip architectures and design methodologies". In Microprocessors and Microsystems- Embedded Hardware Design 35:83-84 2011