Providing QoS to Connection-less Packet-switched NoC by Implementing DiffServ Functionalities

Mehmet Derin Harmanci, Nuria Pazos Escudero, Yusuf Leblebici and Paolo Ienne
Swiss Federal Institute of Technology Lausanne
Processor Architecture and Microelectronic Systems Laboratories
IN-F and EL-D Ecublens, 1015 Lausanne, Switzerland
Email: derin.harmanci/nuria.pazos/yusuf.leblebici/paolo.ienne@epfl.ch

Abstract

QoS is not intrinsic in most of todays NoC solutions, although it is crucial for global predictability and design reuse. Therefore, the current work presents a novel approach that borrows several concepts of DiffServ technology from internet networks and adapts them to NoCs. This novel implementation based on connection-less communication improves the compromise between guaranteeing different traffic requirements and resource utilization, which is not efficiently granted in connection-oriented techniques, and provides a better scalability than the last ones.

1. Introduction

Multi-core systems, widely known as Systems-on-Chip (SoC), have appeared as a promising alternative to face the difficulties of creating even faster uni-processor systems using the latest technologies. They consist of a number of Processing Elements (PEs) and Storage Elements (SEs) connected by a complex on-chip communication architecture (e.g., a Network-on-Chip (NoC)).

The design of flexible, scalable and reliable on-chip communication architectures that meet the constraints and requirements of today's multi-core systems poses many challenging problems. A significant performance degradation of on-chip shared buses due to (1) excessive bus conflicts on single shared buses and (2) bus bridge overhead on hierarchical ones, has driven the research of different alternatives. At present, the so-called Network-on-Chip (NoC) constitutes one of the most promising approaches: It mainly attempts to borrow the concept of packet-switched communication from the area of computer networks and adapt it to design on-chip communication networks. Basic concepts and components of packet-switched networks, such as network topology (e.g., 2-D mesh, hypercube or binary tree), communication schemes (e.g., connection-oriented or connection-less) or communication protocols among others, are being re-examined within the scope of on-chip communication architectures.

One important concern that has not been broadly addressed yet is the importance of providing Quality of Service (QoS) in packet-switched networks for on-chip communication. QoS is characterized by diverse parameters such as availability, delay, jitter, packet loss, and throughput. A study of the existing mechanisms on computer networks and their applicability to this relatively new domain has not been exploited yet. QoS is essential for global predictability and crucial to enable design reuse, but, unfortunately, is not granted in most of the NoCs. They mostly provide Best-Effort (BE) traffic, where all packets are handled in the order they arrive at the system as long as there are sufficient resources available. BE traffic provides a good utilization of communication resources in both Constant Bit-Rate (CBR) and Variable Bit-Rate (VBR) workloads, but, as main drawback, does not meet the requirements of real-time applications. By applying additional services to guarantee traffic characteristics ensures the predefined QoS parameters. This is required by real-time communication and is necessary for easy IP integration. However, resource utilization, specially for connection-oriented schemes running VBR applications, remains relatively low.

The current work presents a novel router architecture for on-chip communication that guarantees different QoS parameters while improving the resource utilization of existing approaches for VBR workloads. It borrows and adapts the IP-DiffServ technology to build a novel packet-switched DiffServ-NoC based on connection-less communication.

The rest of the paper is organized as follows. Section 2 surveys related work. In Section 3 an introduction in DiffServ technology and its application to NoCs is given. Section 4 describes the proposed DiffServ-NoC Scheme. In Section 5 a brief analysis of the implementation complexity is provided. Finally, the results are shown in Section 6 using a multimedia case study (MPEG-2 video decoder).

0-7803-8558-6/04/$20.00 ©2004 IEEE.

2. State of the Art

Although the domain of NoC is relatively young [1], there already exist several works devoted to this topic, both on the theoretical concept and on modelling and implementation issues. For instance, the flexible NoC architecture called Proteo [9] is motivated by the IP design flow and the basic principles behind it. Moreover, it has been demonstrated the feasibility of building on-chip packet-switched communication (SPIN) in the existing technology [4].

Nevertheless, there exist few groups dealing with the importance of providing QoS to the NoC. One of the most relevant approaches in this area is the communication network called Æthereal proposed by Philips Research Laboratory [8]. Such a network supports Guaranteed-Throughput (GT) for real-time applications and Best-Effort (BE) communication where throughput is not guaranteed, but no data is lost [7]. For the GT service they suggest a connection-oriented communication system, which, as main drawback, delivers a non-efficient resource utilization and is therefore not suitable for VBR communication. A further attempt of providing guaranteed bandwidth service has been implemented within the Nostrum NoC architecture [6], reached, as the previous approach, via virtual circuits on a connection-oriented system.

3. DiffServ Technology for QoS on NoC

In order to be able to place and justify the proposed DiffServ-NoC approach, it is first necessary to provide a survey of the different techniques and services supported by today's computer networks and introduce various well-known terms in this field.

3.1. QoS in IP Networks

In order to support diverse applications which have various characteristics and require different service levels, the Internet Engineering Task Force (IETF) has mainly proposed two end-to-end IP service classes [2]: Integrated Services (IntServ) and Differentiated Service (DiffServ).

IntServ are well suited for reliable real-time communication and provide a connection-oriented distinction between flows. Connection-oriented communication is characterized by resource reservation. That is, flows must set up paths through the network and reserve resources at each networking node. Main disadvantages of these systems are a non-efficient resource utilization, the overhead introduced by the connection setup and their non-scalability.

DiffServ provides different levels of QoS to each class by aggregating traffic into different classes at the network edge, and by scheduling packet forwarding for each class within the network. This results on a connection-less communication, which offers a better adaptation of communication to the varying network traffic and a better utilization of network resource, making it suitable to VBR communication (e.g., MPEG). Although connection-less schemes provide a poorer QoS support as connection-oriented techniques, this can be greatly improved by implementing additional services such as DiffServ on top of it.

3.2. DiffServ Technology

DiffServ is based on relative priorities with different sensitivities to delay and loss, but without quantitative guarantees. At the ingress nodes of the DiffServ network packets are classified and marked in the DS-field with the corresponding priority.

The mechanism to allow QoS control at each network node is based on the following elements. A packet classifier is used to classify packets based on the predefined rules. A policer measures the input traffic and assures that the packet behavior follows the predefined profiles. A scheduler controls the packet transmission sequence from queues of individual classes to provide traffic of each class with the QoS level appropriate to the respective class. Last, a queue management accurately controls packets during congestion.

3.3. DiffServ onto NoC

The elements introduced by the DiffServ technology in IP networks can easily be adapted to the NoC paradigm. In this way, the NoC can offer different communication services while providing a better utilization of the network resources, opposite to the poorly resource exploitation for VBR applications of techniques based on connection-oriented communication, such as IntServ.

Nevertheless, it is mandatory that the complexity of the NoC does not increase unnecessarily, introducing much overhead to the on-chip communication. Therefore, the current work has borrowed a subset of the DiffServ elements, which have been considered most relevant, for which a low-cost implementation is envisioned.

The NoC infrastructure mainly contains Processing Elements (PEs) and Storage Elements (SEs) that communicate through a network of routers. These routers act at the same time as edge routers, for incoming/outgoing packets from/to the PEs/SEs, and as core routers, for packets arriving from other routers. In the proposed NoC implementation, each router performs the following tasks on incoming packets: parsing of packet header, classification of the packet, determination of the next hop, queuing, and link scheduling.

4. DiffServ-NoC Scheme

In the following, the conception of the NoC architecture first and its modelling later are explained in detail.

4.1. DiffServ-NoC Architecture

Fig. 1 shows an overview of the scalable NoC infrastructure, composed of a number of homogeneous routers organized following a 2D-mesh topology. A Processing (PE) or Storage (SE) Element is attached to each router (R), which acts as a host sending or receiving packets. There exists a pair of physical links binding two routers or one router with its respective host, one for each direction.

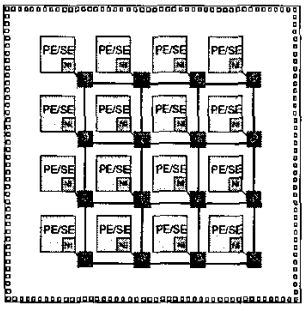

Figure 1. Scalable NoC Infraestructure

4.1.1 Routers

The implementation of each router follows a layered approach by performing the functions related to the lower three layers of the the OSI model (physical, data-link and network layer) [10].

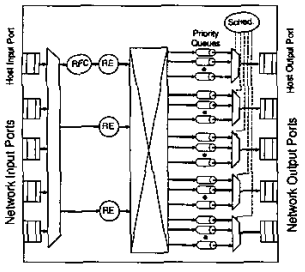

Figure 2. Router Internal Architecture

Fig. 2 shows the generic router architecture of a router (R) in the 2D-mesh topology depicted in Fig. 1. The datapath followed by the packets within the router is the following: Packets are stored in the corresponding input-FIFOs attached to each input physical port. A round-robin mechanism progressively selects a packet for a non-empty input-FIFO and sends it, either to the classification module (RFC) for packets arriving from the host, or directly to the engine processing the routing algorithm (RE). The classification engine sets the DS-field of the packet header to indicate the service the packet should receive within the network. Later on, the routing algorithm calculates the next hop based on the destination address (DA) of the packet relying on a static routing table. Then, a switch network forwards the packet to the selected priority queue of the corresponding output port. Lately, a scheduling module selects the first packet of the highest-priority non-empty queue for each output port and forwards it to the corresponding output-FIFO. The router is able to process three incoming packets in parallel.

4.1.2 Processing and Storage Elements

PEs and SEs act as hosts inside the NoC, sending/receiving data packets to/from the target/source host. For interfacing to the packet switched network they require a wrapper, usually called a network interface (NI), which enables the PE/SE to communicate with the network using packets. This NI implements at least the lower three layers of the OSI model and separates communication from computation.

A NI mainly consist of two parts: A Host-Specific (HS) part and a Host-Independent (HI) part. While the first one depends on the host type, the second part can be reused in every host attached to the same NoC. The HI part is responsible for packetization and depacketization of data and data encoding for error detection and correction.

4.2. DiffServ-NoC Modelling

The system-level language SystemC has been selected for the Transaction-Level Modelling (TLM) of the proposed NoC architecture. Its modularity, its characteristics such as the modelling of time, reactivity and concurrency, and its help in evaluating resource contentions, makes SystemC especially suitable for the modelling and further evaluation of the proposed DiffServ-NoC.

On top of SystemC the On-Chip Communication Network (OCCN [3]) library has been applied, which facilitates the developing of new models for on-chip communication architectures. In the current work, the OCCN library has been mainly used for the definition of the Packet Data Unit (PDU) and the functionalities related to the lower two layers, physical and data-link layer.

5. Complexity Analysis

In the definition of the router and NI architecture for a NoC special care has to be devoted to the storage and computation resources, which are relatively more expensive on-chip than in traditional computer networks. Therefore, for the current implementation, the recursive flow classification (RFC) algorithm proposed by Gupta [5] and a priority-based scheduling algorithm have been selected due to its low-cost implementation. And as buffering strategy an output queuing with 4 priority queues for output port has been applied.

6. Case Study: MPEG-2 Video Decoder

The implementation of multimedia applications, as MPEG-2 video decoding algorithm, on multi-core systems poses many new challenges in terms of inter-core communication. Therefore, it has been used as case study in the current work for demonstrating the necessity of a compromise between guaranteed traffic and resource utilization when defining the NoC.

Fig. 3 shows the functional data flow of the selected video decoding algorithm as well as its partitioning into concurrent tasks, which have variable consumption/production rates, communicating through the implemented DiffServ-NoC.

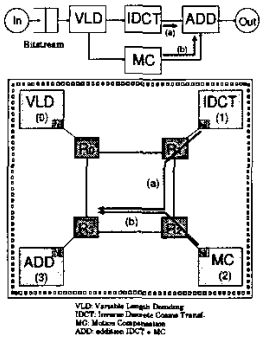

Figure 3. MPEG-2 Video Decoding on NoC

The aim of the system simulation is to demonstrate the possibility of controlling the bandwidth associated to each packet flow by varying the given priorities in the classifier thus providing a defined QoS. This is depicted in Fig. 4 for the data flow (a) from host IDCT to host ADD, which is sent in parallel to the data flow (b) from host MC to host ADD. The bandwidth of the physical link between $R_2$and R_3 is shared by both flows and therefore the end-to-end delays for the packets of one flow vary depending on their priority respect to the other flow. This can be seen in Fig. 4 for the flow (a) having a higher or lower priority than flow (b), respectively. The differences in the end-to-end delay for both simulations and for the same packets (packetID) have been projected to the X-Y plane for observability.

Concerning the resource utilization, it is expected that the proposed approach based on connection-less communication applying DiffServ results on a higher utilization of the available resources than a connection-oriented technique, specially in VBR applications. This effect has been already proven in the area of off-chip networks.

7. Conclusions and Future Work

This work has presented a novel NoC architecture that supports several QoS concepts borrowed from DiffServ techniques. It has been demonstrated that the bandwidth of a shared physical link can be proportionally split between data flows by associating different priorities to each flow, whereas the available resources remain highly utilized.

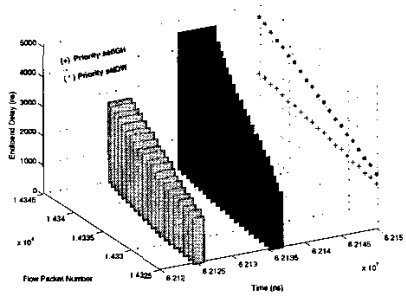

Figure 4. End-to-end Delay of Flow IDCT-ADD (a)

In order not to pay a high penalty in terms of delay or area, since the resource constraints (e.g., storage and computation resources) are tighter on-chip than off-chip, a low-cost implementation of the required resources have been selected for the internal router architecture. The completion of the design flow towards the silicon implementation will be covered in future work.

References

[1] L. Benini and G. D. Micheli. Networks on chips: A new soc paradigm. *IEEE Computer*, pages 70–78, jan 2002.

[2] R. Chakravorty, S. Kar, and P. Farjami. End-to-end internet quality of service (qos): An overview of issues, architectures and frameworks. Bhuwaneshwar, India, dec 2000.

[3] M. Coppola and al. *On-Chip Communication Network: User Manual V1.0.1*. 2003.

[4] P. Guerrier and A. Greiner. A generic architecture for on-chip packet switched interconnections. In *Proceedings of DATE'00*, Paris, march 2000.

[5] P. Gupta and N. McKeown. Packet classification on multiple fields. In *Computer Communication Review of ACM SIGCOMM'99*, Cambridge, Massachusetts, oct 1999.

[6] M. Millberg, E. Nilsson, R. Thid, and A. Jantsch. Guaranteed bandwidth using looped containers in temporally disjoint networks within the nostrum network on chip. In *Proceedings of DATE'04*, Paris, feb 2004.

[7] E. Rijpkema and al. Trade offs in the design of a router with both guaranteed and best-effort services for network on chip. In *Proceedings of DATE'03*, Munich, march 2003.

[8] E. Rijpkema, K. Goossens, and P. Wielage. A router architecture for network on silicon. In *Proceedings of Progress'01, 2nd Workshop on Embedded Systems*, 2001.

[9] I. Saastamoinen, D. Siguenza-Tortosa, and J. Nurmi. An ip-based on-chip packet-switched network. *Book: Networks on Chip*, pages 193–213, 2003.

[10] A. S. Tannenbaum. *Computer Networks*. Prentice Hall, 3rd edition, 1996.