Feasibility study of a core router based on a network on chip

Andreas Ehliar, Daniel Wiklund, and Dake Liu
Dept. of Electrical Engineering
Linköping University
S-581 83 Linköping, Sweden
{ehliar,danwi,dake}@isy.liu.se

Abstract

In this paper we investigate the feasibility of creating a core router based upon a network on chip. The investigated architecture uses 16x10-Gbit Ethernet ports. The purpose of this is to show that it is possible to create such a solution considering current process technologies. This is done through an analysis of the required chip area, clock frequencies, and pin count. The results show that such a solution is feasible and can be implemented as a single chip.

1. Introduction

The steady increase in bandwidth has placed higher and higher demands on all parts of the Internet infrastructure. The backbone of the Internet consists of a core network to which all service providers are connected. The routers in the core network are called core routers whereas routers connected on the edge of the backbone are called edge routers. This is illustrated in figure 1. Edge routers can provide features such as firewalling and quality of service whereas core routers are focused only on packet delivery with no or little capacity left for advanced routing decisions. Current core routers are big and power hungry and the focus of this paper is to determine if it would be possible to fit the functionality of one 16 port core router based on the 10-Gbit Ethernet standard in one chip.

A previous project investigated the feasibility of implementing a core router using a network on chip from the internal communication perspective. This study is based on the SoCBUS network from Linköping University [1]. The results showed that such an implementation was feasible at 16 ports each running at 10 Gbit/s.

The data flow for the router is shown in figure 2. The incoming packets are filtered at the input packet processors. The header information is sent to the forwarding table while the payload is sent to the packet buffer. When the forwarding table has taken a routing decision it will send the information to the packet buffer which will concatenate this with the payload and deliver it to the appropriate output packet processor. The output packet processor is responsible for adding checksums, CRCs, and other miscellaneous tasks before sending the packet to the output line.

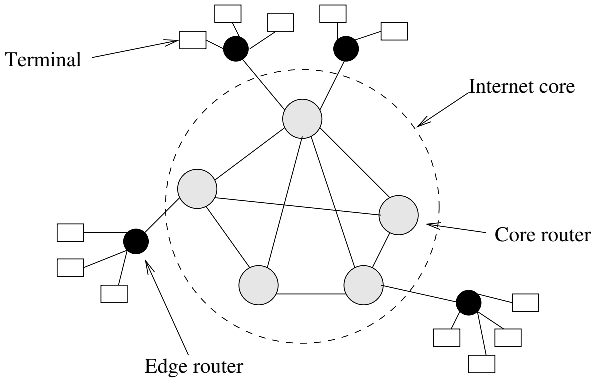

Figure 1. A view of the location of core routers and edge routers in the Internet.

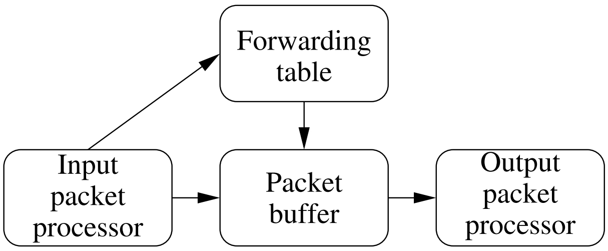

Figure 2. The data flow in the router model.

The rest of this paper is organized as follows: section 2 contains a feasibility study of the various components of the router, section 3 discusses future work, and section 4 contains our conclusions.

2. Feasibility study

In this section we will describe the problems involved in the core router chip. We will focus on the network on chip interconnect, the input packet processor, the packet buffer and the forwarding table.

2.1. Network on chip

The network developed within the SoCBUS project has been used as basis for the basic communication study [2]. This network has a complete toolchain for analysis of the system performance which has been used to estimate the performance of the router.

Models were implemented for the different components used in the router flow. No internal processing is included in the models since they are used mainly for internal traffic modeling. This implies that the models have to handle the traffic flows accurately on a high level.

Traffic models were developed for a set of SoCBUS topologies and allocations and the final result showed that the goal of 16x10 Gbit/s is feasible considering only the internal communications.

Figure 3 shows the final allocation of the router components to the SoCBUS network. The packet buffer must have more than one SoCBUS port in order to meet the bandwidth requirements of this component. This is also true for the forwarding table, although the bandwidth requirements are significantly lower. Additional blocks that are included in the allocation are the control processor (CPU) and the multicast unit (MU). These were not modeled in the study.

The SoCBUS five port routers capable of 1.2 GHz operation occupies at most 0.06 mm^2 each in a 0.18 μm process [3]. Thus, the entire network on chip will occupy less than 3.36 mm^2 excluding wiring.

2.2. Input packet processor

The input packet processor, IPP, is responsible for checking the CRC, validating the IP checksum, and validating the IP header. This input processor has to be able to handle packets coming in at wirespeed. If the processor is built to handle 32 bit data at a time it has to run at 312.5 MHz. Henriksson describes a a protocol processor able to run at 281MHz in a 0.18μm process [4]. Using a state of the art technology in 2005 it is reasonable to assume that this will be able to run at the required 312.5 MHz. Area wise, in 0.18μm, the protocol processor core uses 0.4 mm^2. 16 input processor cores will use 6.4mm^2.

Another question is how the IPP is connected to the Ethernet PHYs. It could use either an XGMII interface or an XAUI interface. The XGMII is a parallel interface and the XAUI is a serial interface. The XGMII interface receive interface consists of 32 data signals, 4 control signals, and 1 clock signal. The transmit interface has the same signals going in the opposite direction. The XAUI interface on the other hand consists of 4 differential pairs in each direction. The XGMII and XAUI specifications are available in the 10 Gbit/s amendment to the Ethernet standard [5].

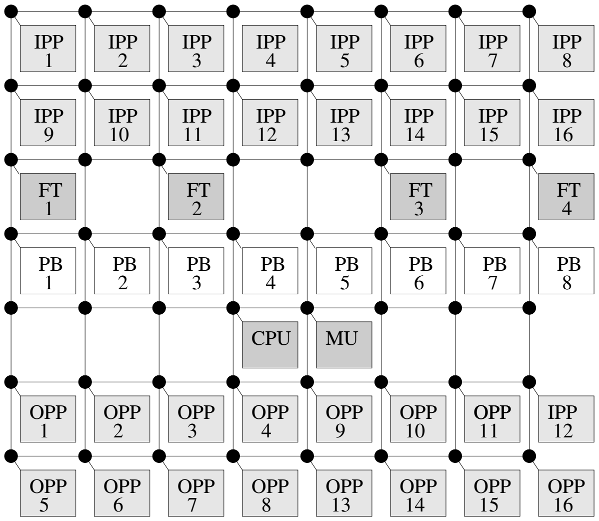

Figure 3. The network topology of the network on chip used in the core router.

The number of pin required for 16 XGMII interfaces is 1184. This would make packaging hard and rules out an XGMII based solution. For the XAUI solution, only 128 pins are used. The drawback is that it is harder to create the interfaces since serializers/deserializers, SERDES, have to be used. A SERDES circuit with 8 channels can be created in 2 mm^2 in a 0.16 μm technology [6].

2.3. Packet buffer

The packet buffer is a big challenge. The main property to be taken into account while designing the packet buffer is the size of the buffer. Put in another way, assuming that the router is, on average, getting packets evenly distributed to all output ports, how long time should it be able to handle an overload situation where more than 10 Gbit/s is destined for the same output port? In a situation where packets on all ports are destined for one output port, a buffer capable of buffering 0.1s of traffic will be 2 GB large. This is clearly not feasible to have on-chip with current technologies.

As even a single lost packet dramatically reduce TCP performance there is a need for a large off-chip buffer to avoid packet drops in case of a temporary congestion. In this paper we discuss RDRAM for the off-chip buffer. The advantage of RDRAM over DDR-SDRAM is that the pin count of RDRAM is smaller for the same amount of bandwidth. The I/O pin count for one RDRAM module is 34 pins.

The required bandwidth to the off-chip memory buffer is 40GB/s. At a clock frequency of 1600 MHz, this will give a sustained data transfer rate up to 3.2 GB/s per RDRAM module [7]. 13 such modules are required to get a bandwidth of over 40 GB/s for a total pin count of 442 pins. If the 512 Mb module is used this will amount to a total of 832 MB of memory and the router will be able to handle a total of 44 ms worst case traffic before dropping packets.

The size of a Rambus ASIC cell has been reported to be 3.6 x 1.1 mm^2 in a 0.25 μm technology [8]. The size of 13 such cells in 0.25 μm technology is 52 mm^2.

2.4. Forwarding table

The task of the forwarding table is to look at the destination address of a packet and decide which port the packet should be sent to. In order to determine the feasibility of implementing the forwarding table on chip the following properties must be taken into account:

- Number of prefixes in forwarding table
- Chip area
- Throughput
- Latency

The number of prefixes in the forwarding table is determined by the routing tables used on the Internet. As of the beginning of 2005, the number of active (IPv4) BGP entries is close to 200000 entries [9]. In order to be future safe for a couple of years, we decide that the core router should be able to handle 300000 entries.

Each prefix in an IPv4 forwarding table consists of an entry for a network address, an entry for the number of significant bit in the network address, and an entry for the address of the next hop. The task of the forwarding table is to find a matching address with the highest number of significant bits in the network address. This is called longest prefix match. Ruiz-Sanchez has a survey of available algorithms [10]. For this paper we will use range search as an example.

The memory usage of range search is (worst case) twice the number of bits in the prefix address and twice the number of bits used to identify the destination. In a very simple router the destination address would be the same as the destination port plus a flag to indicate that the packet should be dropped. However, to make it possible to create more advanced routing tables we will consider a forwarding table where we can have 2^{12} possible destination classes. The storage requirements for the forwarding table would then occupy a maximum of $300000 \cdot 2(32 + 12) = 26.4$ Mbit.

According to the International Technology Roadmap for Semiconductors it will be possible to fit 84 million SRAM cells in one cm^2 in 2005 [11]. Allowing for the overhead incurred by the surrounding logic, the forwarding table will still fit into 0.5 cm^2.

The latency measured in clock cycles is dependent upon how pipelined the forwarding table is. Assuming that the range search is implemented by a series of memories where each memory is twice as large as the previous one, 19 memories will be required. At three pipeline stages per memory, the latency of the forwarding table will be 57 clock cycles. In this architecture the forwarding table can issue one search request per clock cycle.

As for throughput, the maximum number of search requests per second is equivalent to the maximum number of packets per second. For 10 Gb Ethernet this amounts to approximately 16.5 million packets per second and port. Sixteen ports can produce 264 million packets per second. If the forwarding table is clocked at around 300 MHz it will be able to handle all lookup requests while still having free cycles left for table updates.

The above discussion is valid for IPv4. As for IPv6, this should not be a problem since one of the main goals of IPv6 is that the large address space of IPv6 will be used to create an hierarchical address space with dramatically reduced routing tables as a result.

2.5. Other components

The chip should contain some sort of microprocessor to deal with management of the router like updating the forwarding table. The performance of the microprocessor is not going to be critical. An ARM1026EJ-S occupies a chip area of 4.2 mm^2 including caches in a 0.13 μm technology [12].

2.6. Feasibility summary

The chip area and pin counts for the components are summarized in table I. It is important to keep in mind that these figures are pessimistic since most of the referenced papers used non-current technologies. In particular, we expect that the 52 mm^2 for the RDRAM interface is very pessimistic but we have not been able to find any figures for a technology newer than 0.25 μm.

Some components have not been analyzed in detail, e.g. the RDRAM controller and the on-chip packet buffer. These components will be investigated as the need arise. These are not expected to be prohibitively large. According to the ITRS report, the chip size of a microprocessor at introduction can be 280 mm^2. The proposed router architecture is well within these limits.

Table I. Area estimates for the different components of the core router

Component	Chip area	Pin count
Network on chip	3.36 mm^2	-
Input packet processor	6.4 mm^2	-
XAUI interfaces	16 mm^2	128
RDRAM interface	52 mm^2	442
Forwarding table	50 mm^2	-
Microprocessor	4.2 mm^2	-

3. Future work

We are currently planning to build a demonstrator on an FPGA prototype board with up to eight 1 Gbit/s Ethernet ports. The intention is to show that SoCBUS, the route lookup engine, and the protocol processor can be used in the same system.

4. Conclusions

This paper shows that it is feasible to implement a core router with 16x10-Gbit Ethernet ports as a single chip with current state of the art technologies. External memory is required in order to handle temporary congestion situations.

References

[1] D. Wiklund and D. Liu, "Design of a system-on-chip switched network and its design support", *Communications, Circuits and Systems and West Sino Expositions, IEEE 2002 International Conference on*, vol. 2, n. 29, 2002.

[2] J. Svensson, "Design of a core router using the SoCBUS on-chip network", Master's thesis, Linköping University, 2004.

[3] S. Sathe, D. Wiklund and D. Liu, "Design of a switching node (router) for on-chip networks", *ASIC, 2003. Proceedings. 5th International Conference on*, vol. 1, 2003.

[4] T. Henriksson, *Intra-Packet Data-Flow Protocol Processor*, PhD thesis, Linköping University, May 2003.

[5] IEEE, *IEEE Standard for Information technology- Telecommunications and information exchange between systems- Local and metropolitan area networks- Specific requirements Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications Amendment: Media Access Control (MAC) Parameters, Physical Layers, and Management Parameters for 10 Gb/s Operation*, 2002.

[6] Fuji Yang, J. O'Neill, P. Larsson, D. Inglis and J. Othmer, "A 1.5V 86mW/ch 8-channel 622-3125Mb/s/ch CMOS serdes macrocell with selectable MUX/DEMUX ratio", *Solid-State Circuits Conference, 2002. Digest of Technical Papers. ISSCC. 2002 IEEE International*, vol. 2, 2002.

[7] Rambus, "RDRAM 512Mb (1024Kx16/18x32s)", on the www, http://www.rambus.com/products/rdram/documentation/rdram.512s.0205-03.pdf.

[8] K. et al Suzuki, "A 2000-MOPS embedded RISC processor with a Rambus DRAM controller", *Solid-State Circuits, IEEE Journal of*, vol. 34, n. 7, pp. 1010–1021, Jul 1999.

[9] G. Huston, "BGP Routing Table Analysis Reports", on the www, http://bgp.potaroo.net/.

[10] M.A. Ruiz-Sanchez, E.W. Biersack and W. Dabbous, "Survey and taxonomy of IP address lookup algorithms", *Network, IEEE*, vol. 15, n. 2, pp. 8–23, 2001.

[11] International Technology Roadmap for Semiconductors, "2004 Update Overall Roadmap Technology Characteristics", on the www, 2004, http://www.itrs.net/Common/2004Update/2004_000_ORTC.pdf.

[12] ARM, "ARM1026EJ-S", on the www, http://www.arm.com/products/CPUs/ARM1026EJS.html.