2004 IEEE INTERNATIONAL CONFERENCE ON FIELD-PROGRAMMABLE TECHNOLOGY: PROCEEDINGS pdf

1: IEEE EDOC Conferences

Proceedings. IEEE International Conference on Field- Programmable Technology (IEEE Cat. NoEX).

Shared buffer architectures consequently become one of the dominating constructs of modem routers and switches. This paper investigates new and existing shared buffer architectures that are ideal for emerging FPGA technologies with embedded memory. Introduction The widespread availability of broadband Intemet is now affecting every aspect of daily life. As people increasingly rely on the new services for their lifestyle and work. Quality of service QoS is therefore likely to become a major differentiator in a highly competitive telecommunications market. The support for application specific QoS from communications networks is also a key requirement. Packets are usually transmitted through the network based on their destination address, attached as one of the packet header fields. In comparison to traditional circuit switched networks, where user data is transmitted through a preconfigured route, in packet switched networks packets are stored and examined before being forwarded to the subsequent network node. The asynchronous nature of packet based communication postulates, relative to classical circuit switching, that large amounts of memory are required for the storing of arriving packets. Subsequently, packet contention is expected, when more than one packet from N input ports are addressed to a particular output port. Packet contention at output ports is resolved by utilising a packet service policy. The increase of throughput rate and the demand for service variety requires that more efficient packet buffers and faster packet retrieval techniques are employed. This demand can not be satisfied by increasing the physical memory size of the network nodes. Provisioning of QoS also means the provisioning of packet buffers for individual flows. Sharing resources, in particular the packet buffer resource at network nodes, are essential for the economic operation of sophisticated QoS aware routers and switches. This paper examines advanced shared buffer architectures that are not only economic from the resource sharing point of view, but also provide scalability and fast packet-retrieval to meet the requirements of both interactive and multimedia services. It is anticipated that on-chip embedded memory, provided by platforms such as FPGAs and System on Chip SoC technology, will significantly advance the retrieval and access of packets stored in external shared memory. Novel shared buffer architectures based on FPGA technology are investigated to overcome the limitations of classical shared buffer architectures proposed and deployed for packet switching, routing and scheduling. In particular, packet caching using embedded memory is expected to speed up the effective throughput rate of classical shared buffers and is one of the main objectives of this research. Related work The concept of sharing packet buffer resources evolved from the principles of shared memory, were a common memory resource is shared among multiple processors. The renewed significance of shared buffer architectures began with the emergence of the ATM network and ATM switching requirements. Hitachi Labs developed one of the fust shared buffer based switches, using custom LSI technology [l]. Many of the latest shared buffer architectures are based on the same approaches used by Hitachi. CNET introduced their Prelude switch in [2]. The Roxanne architecture was proposed in It has been developed and introduced by Alcatel research labs [3]. Architectures have been implemented that provide scalability in buffer size, since queue address control is independent fiom buffer memory control [SI. The presented architecture is capable of operating at 80 MHz and has a throughput of 2. Numerous architectures have been proposed addressing the scalability issues of shared buffer architectures [9], [IO], [Ill. All of these approaches are based on multi-buffer architectures. The sharing principles are based on the concept of the rearrangeable non-blocking multistage interconnection networks proposed by Clos [I21 and Benes [13]. Several architectures have been developed to improve the memory access time of large shared buffers [14, These architectures have divided a large memory buffer into smaller multi-buffer banks. They have solved the problem of the buffer access time restriction but required high speed switching circuits and complex address control. Those architectures cannot provide the scalability of shared buffer for flexible switching performance, as memory bank control is not independent of address control. The resultant switching speed is dramatically increased as there are multiple

blocks and the actual data path to a specific memory cell is reduced. The resulting RAM array configuration is scalable for larger memory sizes. A pipelined architecture is introduced in SRAM to support the fast access speed of a single memory block. The scalability is the constant memory cycle time, regardless of the buffer size. Recent research on shared buffer architectures targets memory efficient solutions for variable sized packets. One of the most promising solutions is presented by Michael V. The shared buffer architecture is based on the tail buffer concept, where the tail of a packet that is below a given threshold size is stored to a separate tail buffer, instead of the main shared buffer. Memory utilization is therefore improved. Buffer management is a key consideration when implementing shared buffer architecture. Separate FIFO queues may become oversized for large shared buffers. Embedding pointer data [I91 in a contiguous area inside the packet memory is used to effectively implement a circular queue for packets. I External RAM 3. Sharing packet buffer resources In regular shared buffer architectures Figure 1 packets are shared in a common memory when they arrive and the packet header is extracted and routed to the output port. When the output port scheduler schedules a packet for transmission, it removes the packet fiom the shared memory. It can be seen in Figure 1 that the idle buffer queue controls the fiee memory locations for the packets. Once the packet has been scheduled the fiee memory address is fed back via the idle buffer queue, hence allowing a new packet to be stored at the given address. Shared buffer architecture Any shared buffer architecture is limited by its medium access. An N port shared buffer must he capable of processing N incoming and N outgoing ports. Investigated shared buffer architecture This initial research has addressed the architectural issues that are associated with shared buffers. Two architectures are investigated and implemented in VHDL. Firstly, an architecture is developed whereby for each incoming packet, a fixed proportion of the memory is allocated. A pointer, addressing the physical memory location that accommodates the packet, is generated and stored in a service FIFO. The pointers are then processed sequentially, in accordance with their arrival sequence. For simplification the FCFS scheduling policy has been adopted. The author would like to emphasise that the investigated shared buffer architectures are independent from the scheduling policy and that more complex polices, such as WRR or WFQ can he deployed. A generic pointer FIFO has been implemented to accommodate address pointer handling. A FIFO allows a fastest retrieval of address pointers and the most efficient way of utilising the shared memory bandwidth for payload data. The first architecture was based memory segmentation into block of equal sizes. It should he noted that this architecture is only suitable for fxed packet sizes and is therefore restricted. The second design Figure 2 was subsequently developed ftom the first architecture. The allocation of variable buffer space has been achieved by concatenating of multiple memory blocks. It is the function of the extemal memory controller to handle the address locations of the packets. The internal address pointer FIFOs are based on dual port embedded memory. In order to improve the resource utilisation the extemal shared memory is distributed into m byte blocks. Multiple blocks of bytes are used to accommodate one packet of several kilobytes. The first eight bits are assigned from counter values, which correspond to the block addresses within the extemal RAM. Bits 19 and 20 are the packet status control information. Pointer formatting The packet status control information indicates whether the memory block belongs to the start, middle or end of a stored packet. This enables variable packet sizes to be input into the system hy allowing multiple blocks to be concatenated together. The packet status control information will be generated each time the byte count reaches The remaining 12 bits corresponds to the free memory locations in the RAM structure. The system is therefore capable of supporting memory blocks. The designs were synthesised using Synplicity and Xilinx foundation tools. The post layout synthesis results are presented in Table 2. The use of embedded memory presents a significant speed up. For extremely large shared memory architectures, beyond 4 Gbytes, it is expected that the traditional approach will not be sufficient at sharing the pointers on the FPGA. The caching of data and address pointers is expected to resolve the issues of implementing extremely large shared buffer architechres. Nevertheless, the presented concept is ideal for implementing shared buffer based network link modules with the demand for sophisticated schedulmg policies. Conclusion An investigation of current shared buffer architectures has been presented. This paper has

highlighted the need for an improvement in the existing solutions if current systems are to meet future network requirements. Based on a comprehensive literature study, the current bottleneck in shared buffer architectures is found to be the access speed to the external memory. In order to overcome the 2N access speed l i t a t ion , it is suggested that advanced packet caching, utilising embedded memory withim fnture SoCiSoPC architectures, will significantly reduce the need for accessing the external memory. This will therefore have the overall effect of speeding up the throughput rate of a given system. The current investigation presented the preliminary research on shared buffer architectures. The fust implementation is capable of processing fixed and variable sized packets from multiple FIFO From the results in Table 2, it can be seen that an overall operational speed of M H z can be achieved This is implemented using register bits and LUTs, which is well withiin the capacity of existing FPGAs. The investigation has also proven that the latest FPGAs technologies, with embedded block memory, offer an ideal platform for high performance shared buffer architectures. Methods of increasing the memory bandwidth, such as those studied in [20], will also be investigated. Journal on Computer nemorks andlSND systems. Proceedings of h Conference on Local Computer Nehvorks. Phoenix, Arizona , United States, www.

2004 IEEE INTERNATIONAL CONFERENCE ON FIELD-PROGRAMMABLE TECHNOLOGY: PROCEEDINGS pdf

2: Publications & References â€" Verilog-to-Routing dev documentation

Published in: Field-Programmable Technology, Proceedings. IEEE International Conference on Proceedings. IEEE International Conference on Article #.

Xilinx Inc, ug edition, feb Cluster-based logic blocks for fpgas: In Custom Integrated Circuits Conference, â€" PhD thesis, University of Toronto, Directional bias and non-uniformity in fpga global routing architectures. On biased and non-uniform global routing architectures and cad tools for fpgas. Automatic generation of fpga routing architectures from high-level descriptions. Kluwer Academic Publishers, mar Kluwer Academic Publishers, Universal switch modules for fpga design. Efficient fpga mapping using priority cuts. Slack allocation and routing to improve fpga timing while repairing short-path violations. Very Large Scale Integr. Odin ii-an open-source verilog hdl synthesis tool for cad research. Activity estimation for field-programmable gate arrays. Direction and single-driver wires in fpga interconnect. Architecture description and packing for logic blocks with hierarchy, modes and complex interconnect. Using cluster-based logic blocks and timing-driven packing to improve fpga speed and density. Timing-driven placement for fpgas. Cad tools and architectures for improved fpga interconnect. Benchmarking method and designs targeting logic synthesis for fpgas. Kent, Peter Jamieson, and Jason Anderson. Bounds on net delays for vlsi circuits. Analog and Digital Signal Processing, 39

3: Publications - HPGCWiki

Get this from a library! Proceedings / IEEE International Conference on Field-Programmable Technology, [International Conference on Field Programmable Technology (3, , Brisbane); Electron Devices Society; School of Information Technology and Electrical Engineering;].

4: Self-recovery experiments in extreme environments using a field programmable transistor array

Get this from a library! IEEE International Conference on Field-Programmable Technology: proceedings: December , , the University of Queensland, Brisbane, Australia.

5: International Conference on Field Programmable Technology

[IEEE IEEE International Conference on Field- Programmable Technology - Brisbane, NSW, Australia (Dec.)] Proceedings. IEEE International Conference on Field- Programmable Technology (IEEE Cat.

6: CSDL - Conferences Beginning with the letter 'F'

Oliver Diessel, John Williams: Proceedings of the IEEE International Conference on Field-Programmable Technology, Brisbane, Australia, December ,

7: dblp: IEEE International Conference on Field-Programmable Technology (FPT)

Proceedings of the IEEE International Conference on Field-Programmable Technology, FPT , Hong Kong, China, December , IEEE , ISBN [contents] maintained by at.

8: dblp: IEEE International Conference on Field-Programmable Technology (FPT)

2004 IEEE INTERNATIONAL CONFERENCE ON FIELD-PROGRAMMABLE TECHNOLOGY: PROCEEDINGS pdf

ABN 63 International Conference on Field-Programmable Technology (FPT) The University of Queensland St Lucia Campus, Brisbane, Australia.

9: Recent Publications | Electrical and Computer Engineering | UBC

IEEE International Conference on Mobile Data Management, Proceedings. ().

2004 IEEE INTERNATIONAL CONFERENCE ON FIELD-PROGRAMMABLE TECHNOLOGY: PROCEEDINGS pdf

3 Radio Set AN/TRC-7, simplified block diagram with remote control equipment 2 Gervase Sacheverill Ceramic materials P2 nucleotide receptors Josiah Royce in focus Postmortem collectibles Ms access 2007 guide Biomedical sensors Robotic exploration of the solar system part 3 Wallace and Gromit and the Lost Slipper (Wallace Gromit) Envisioning dance on film and video SuzanneS Surrender Electromagnetic theory of propagation interference and diffraction of light Why the Negro is Black On the Great Bahama Bank Treasure of Qumran Marketing management marshall johnston Act II. Innovation How modern interrogators have adapted to Miranda They Called Me Mayer July Polaris 280 parts diagram Mathematical applications for combat modeling P.T. Hester A. Collins A Shadow of My Own The mediation of Christ Toyota spare parts price list Power Tools for Positive Living The Offbeat Kings Indian Pt. 2. Revelations about marriage V. 1. From the year 1783 to the year 1804 The proper care and feeding of church volunteers William Caxtons prologues and epilogues. Strategies for exposing domestic violence in country music videos by Julie Haynes The chocolate rose laura florand Valentine to a flying mouse Voiding disorders A Monster Named Criney Who Makes Kids Whiney Thats What Friends Are for (Changing Picture Books) Sbi money back plan Hipster business models CYBEX INTERNATIONAL, INC.