Hardware Accelerated Imaging Algorithms

Zemcik P., Fucik O., et al: Faculty of Information Technology, Brno University of Technology

The contribution shows some new approaches to hardware acceleration of imaging algorithms using combined FPGA and DSP. The novel approach to hardware acceleration focuses on raster based techniques, such as volume rendering and image processing. The paper describes the basic principles of the architecture and mehods of its usage in computer imaging.

Obrazové algoritmy s akcelerací hardware

Zemèík P. Fuèík. O, a spol: Fakulta informaèních technologií, VUT v Brnì

Pøíspìvek popisuje nový pøístup k akcelerací obrazových algoritmù pomocí hardware s použitím FPGA a DSP. Nový pøístup k akceleraci pomocí hardware se zamìøuje zejm éna na algoritmy pracující s rastry, jako jsou napøíklad "volume rendering" a zpracování obrazu. Èlánek popisuje základní princip nové architektury a obsahuje též metody použit í této architektury ve zpracování rastrového obrazu.

1 Introduction

Image processing is traditionally one of the areas with very high demands for computational power. Such demand is generally a result of a simple fact that the algorithms that the imaging algorithms mostly work with large data sets and the requested processing time is in many cases quite short [1][2].

Applications areas that require such short processing time are e.g. industrial visual quality inspection, real-time image tracking, video processing, compression, and decompression, biomedical applications, and many others. As the algorithm research is progressing, the demand for computational power generally increases.

Historically, the computer imaging algorithms were mostly implemented using software so that the hardware was minimised. At the same time the higher end imaging systems were built with single purpose accelerators.

Today, if higher computational power is needed, it can be achieved by pure clock speedup (simplest method, but it has technological limits), by improving architectures of the system, or by using of parallelism (but parallelism is difficult to exploit from the developer’s point of view). Out of these three possibilities, the architectural changes are the most promising.

A reasonable way forward in architectural changes is being offered by the recent development of Field Programmable Gate Arrays (FPGAs) and Digital Signal processors (DSP) [3][4][5][6][7]. Current technological progress allows implementation of very complex devices in the programmable logic devices and achieves good results even with architectures and algorithms that were previously unthinkable to implement in hardware.

2 Experimental Architectures

At Brno University of Technology, we are carrying out experiments with hardware accelerated graphics and image processing for several years. Our experiments include simulation of image processing algorithms, simulations of graphics algorithms, and implementations of the accelerated algorithms including their application in the industry. We have so far implemented two designs and new designs are under development. All of the designs are intended for real applications as well as research.

The general design goal of all of the systems was to develop modules that would be capable of high performance operation independently of the host PC or connected to a host PC. Important design goal was also to maximize the performance/price ratio rather than achieve the maximum possible performance at any price. Other goal was to make the modules scalable in that sense that it would be possible to interconnect them through high-capacity communication links. All of the designs do contain flash permanent memory and communication devices, that are not that important for the actual operation of the modules and therefore they are not indicated in the schematics, but they has a critical role in starting the systems.

In the first of our designs – Xilinx Spartan and Texas Instruments TMS320C32 DSP – we were testing the basic ideas of the FPGA and DSP co operation. The FPGA in this design served as input/output module for the DSP and also as a processing unit for the bulk data. Due to limitations in computational speed and memory, the module was only used for image processing applications and 2D and wireframe graphics. The design was finished in 1997 (see Figures 1 and 2) [8][9][10].

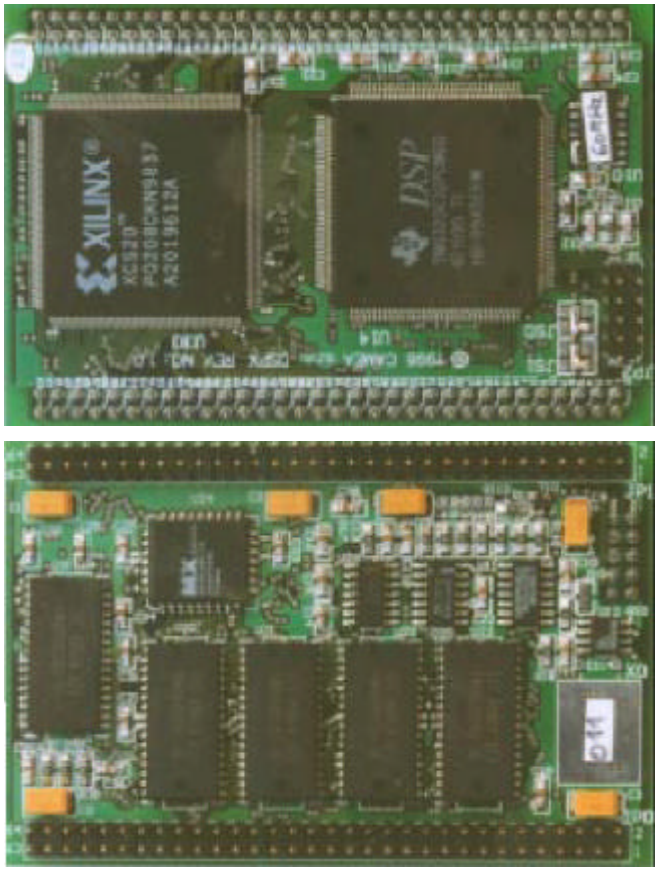

Figure 1: Photographs of the module

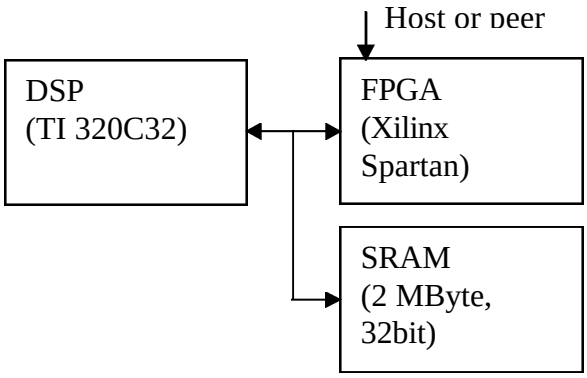

Figure 2: Design with C32 and Spartan FPGA

The design does have some limitations that were forced in the design because the design goal was to achieve the best possible performance/price ratio rather than the best possible performance. Even though, the system is capable of running raster algorithms with significant speedup over the pure DSP systems.

The current generation of the designs was done using the Xilinx Virtex FPGA and Texas Instruments TMS320C6711 DSP. The design goal of the system was to develop a system capable of the high performance image processing and 3D graphics acceleration.

The main differences between the first and the current design is that the new design contains a DSP comparable in performance with the PCs, significantly better FPGA, more memory, and the FPGA can have a small local memory. The main effect of the new design on applications, however, is that the new design supports dynamic reconfiguration of the portions of the FPGA so that the FPGA can swap configurations in the real time and thus maximize the exploitation of the FPGA circuits. The local memory connected to the FPGA allows for local storage of the intermediate results or constant tables, microcode, etc.

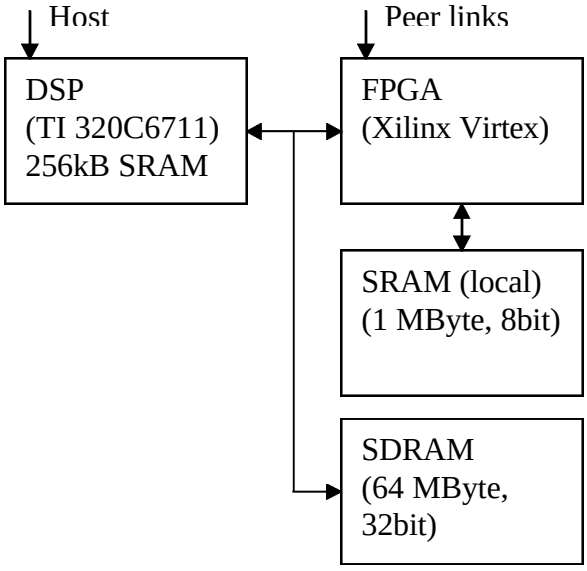

Figure 3: Design with C6711 and Virtex FPGA

The design with C6711 was finished in 2001 and currently we are developing the software support tools for the module. The software is based on the Texas Instruments DSP/BIOS multithreading core and channel-based communication subsystem for data exchange and synchronization between the threads and modules.

Connection of the module to the PC is planned through the PCI motherboard (that is currently being manufactures) that will also contain bulk memory shared among the Xilinx Virtex-E FPGA and the DSP module. The intention of the board is to provide a fast interface between the PC and one or more DSP modules connected to the board directly or through the LVDS links. The memory is intended primarily as a mean of buffering the data.

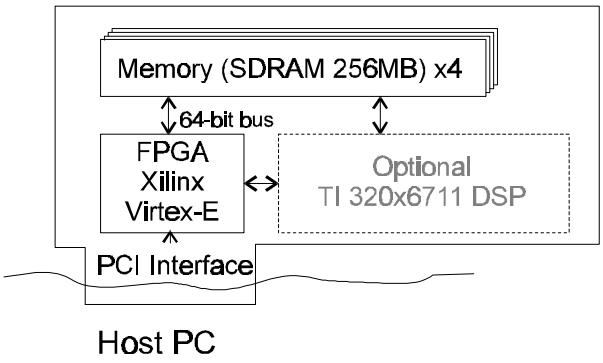

Figure 4: PCI motherboard design

For the future we inted to implement a data processing board using purely FPGA technology without a DSP contained directly in the design as a new series of FPGA with (one or more instances of) embedded PowerPC core. The board will contain up to 1GByte of dynamic bulk memory to allow processing of large data sets for volume rendering, 2 blocks of high-speed memory for temporary data (such as image memory), and a small associative memory to allow caching functions or associative searches for temporarily abandoned processing requests, e.g. in ray casting algorithms.

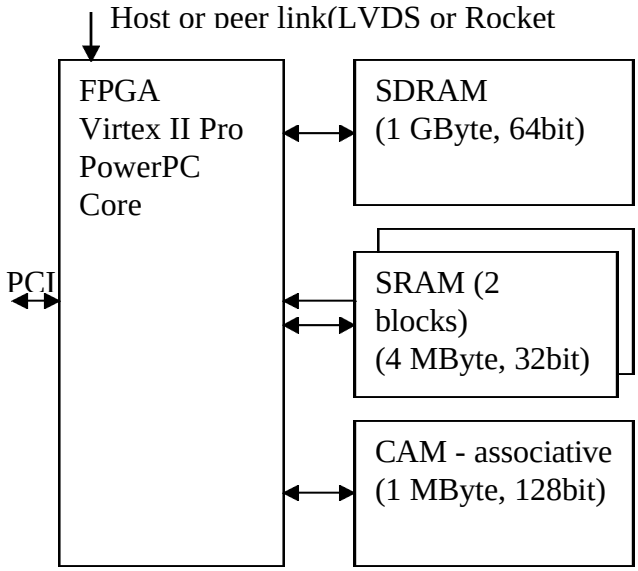

Figure 5: Future design Virtex II Pro FPGA

It is planned that the new design will be finished by the end of 2002.

3 Imaging Experiments

The imaging experiments were generally done using the scheme shown in Figure 6. The best possible situation occurs when data flow is simple and predictable so that the data can be transferred in large blocks using DMA controllers built in the DSP. In that case the algorithms can be run without any assistance of the processor (DSP). The data in such case is latched in the input register RIN, then run through the (optional) buffer system and delivered in a processing unit. The results of the algorithms are transferred back to he output register (ROUT) through the buffer system. Then the data is transferred using the second DMA controller to the system memory. When the algorithm is applied on all the data, the DMA controller (DMA2) issues an interrupt to the processor (DSP) that handles the processing further. This principle must be modified if the data flow is more complex – generally the DSP in that case must assist the data transfers by reprogramming the DMA controllers.

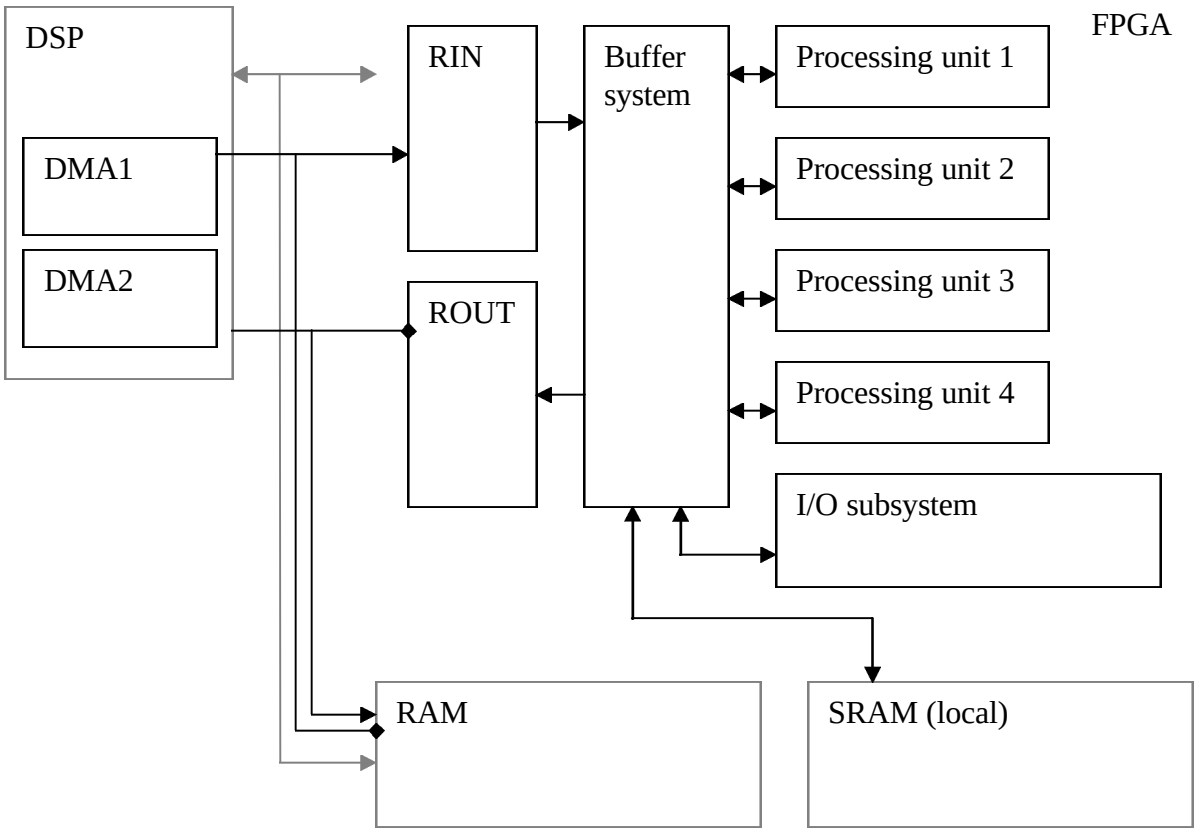

Figure 6: Image processing using the DSP and FPGA

The block called „Processing unit n“ implements the actual imaging algorithms, such as 3x3 convolution filter with fixed coefficients, morphological filters, single pixel functions, such as thresholding, integral imaging functions, etc.

The above described principle is applied on all the designs described in the previous part. All designs but the first one (C32) are capable of partial reconfiguration of the FPGA and have more DMA controllers available. As a result of these facts, the more advanced designs can dynamically allocate the processing units and also can run more processing units at the same time – due to increased maximum throughput of the memory system.

4 Conclusion

The contribution has presented new architectures for acceleration of the imaging algorithms using FPGA and DSP chips. It has been shown that reasonably simple designs using the FPGA can lead in systems useable both in real industrial applications and for research.

The designs shown in the paper have been used in several really running applications and they have been proved to have better performance/price ration than standard designs with e.g. pure DSP solution. (Part of the savings, however, were achieved by using the FPGAs as I/O circuits rather than acceleration of the imaging algorithms.)

Acknowledgements

We would like to thank for the support to all my colleagues, especially Sobeslav Valach and Josef Honec, who helped in the design of the experimental boards and to Pavel Valenta and Miloslav Richter who helped in the application development. Thanks also to CAMEA spol s r.o. that has supplied the experimental material.

The work has been supported by grant of the Grant Agency of the Czech Republic (GAÈR) GA102/02/0507 “Computer Graphics Algorithms with FPGA Support” and by CAK (Centre for Applied Cybernetics – Centrum aplikované kybernetiky) MŠMT.

References

[1] A. Watt: *3D Computer Graphics*, Addison-Wesley, Wokingham, United Kingdom, 1993

[2] B. Jahne, H. Haussecker, P. Geissler: *Handbook of Computer Vision and Applications*, Academic Press, San Diego, CA, USA, 1999

[3] *Spartan and Spartan-XL Families Field Programmable Gate Arrays*, Xilinx, DS060 (v1.6) USA, September 19, 2001, (available at http://www.xilinx.com)

[4] *Virtex™ 2.5 V Field Programmable Gate Arrays*, Xilinx, DS003-1 (v2.5), USA, April 2, 2001, (available at http://www.xilinx.com)

[5] *Virtex-II Pro™ Platform FPGAs: Introduction and Overview*, Xilinx, DS083-1 (v1.0), USA, January 31, 2002 (available at http://www.xilinx.com)

[6] *TMS320C32 Digital Signal Processor*, Texas Instruments, SPRS027C, USA, September 2001, (available at http://www.ti.com)

[7] *TMS3B0C6711, TMS320C6711B Floating point Digital Signal Processors*, Texas Instruments, SPRS088B, USA, September 2001, (available at http://www.ti.com)

[8] O. Fucik, P. Zemcik, R. M. Fors, D. Loker, R. Weissbach: *A Flexible DSP Hardware Platform for LVDT Signal Conditioning*, In Proceedings of Electromotion International Symposium, p. 8, Bologna, Italy, 2001

[9] A. Herout, P. Zemcik: *Raster Volume Data Graphics Library*, In Proceedings ASIS 2001, p. 15, Brno, MARQ, Czech Republic, 2001

[10] P. Zemcik, O. Fucik, M. Richter, P. Valenta: *Imaging Algorithm Speedup Using Co-Design*, In Summaries Volume Process Control 01, p. 96-97, Strbske Pleso, Slovakia, 2001

[11] A. Kaarna, P. Zemcik, H. Kalviainen, J. Parkkinen: *Compression of Multispectral Remote Sensing Images Using Clustering and Spectral Reduction*, IEEE Transactions on Geoscience and Remote Sensing, p. 1073-1082, Piscataway, NJ, USA, 2000

[12] P. Zemcik, P. Valenta, O. Fucik, J. Honec. M. Richter: *Visual Analysis on the Production Line, In Proceedings of The 10th Scandinavian Conference on Image Analysis*, Lappeenranta, p. 301-308, Finland, 1997

[13] P. Zemcik O. Fucik, J. Honec, P. Valenta: *Cost Efficient Image Digitising and Processing Using FPGAs*, In Proceedings Microcomputer'94, p. 225-229, Zakopane, Poland, 1994

[14] A, Kaufman, R. Bakalash: *Memory and Processing Architecture for 3D Voxel-based Imagery*, IEEE Computer Graphics and Applications, vol. 8, no. 6, p. 10-23, USA, November 1988

[15] F. Dachille, A. Kaufman: GI-Cube: *An Architecture for Volumetric Global Illumination and Rendering*, In Proceedings of the SIGGRAPH / Eurographics Workshop on Graphics Hardware. P. 119-128, August 2000

Pavel Zemèík (Doc. Dr. Ing.), Otto Fuèík (Dr. Ing.), Fakulta informaèních technologií VUT v Brnì, Božetì chova 2, 612 66 Brno, e-mail: {zemcik|fucik}@fit.vutbr.cz, tel: +420 5 4114 {1217|1212}, +420 {603 487824|604 206417}