Modular & Scalable Ultrasound Platform with GPU Processing

M. Lewandowski, M. Walczak, B. Witek, P. Kulesza, K. Sielewicz
Department of Ultrasound
Institute of Fundamental Technological Research PAS
Warsaw, Poland
mlew@ippt.pan.pl

***Abstract*—The objective of our project is to develop a complete ultrasound platform with real-time GPU processing. The platform is designed to be modular and scalable both in number of ultrasound channels (64-256), as well as in communication bandwidth and processing power. By standardizing on the PCIe switch fabric, we are planning to integrate all the ultrasound modules and processing resources (GPU) in a single rack enclosure. Using PCIe direct peer-to-peer communication for transferring the data from the ultrasound acquisition modules to the GPUs, we maximize the system bandwidth and minimize CPU usage. The first developed module of our platform is RX64 - a 64-channel ultrasound acquisition PCIe card. The RX64 contains a high-end FPGA Altera Stratix IV 70 GX interfaced to: two 32-channels mixed-signal front-end ultrasound modules and two 64-bit 8GB DDR3 SO-DIMM memories for data buffering. We also develop GPU kernels for SAFT based ultrasound imaging, as well as GPU Framework for building complete signal processing pipeline.**

Keywords— ultrasonic imaging; synthetic aperture; GPU; FPGA

I. INTRODUCTION

A development of ultrasound imaging methods requires an access to raw echo signals and processing of them. The commercial medical ultrasound devices can be optionally equipped with a research package allowing storage of digital signals after and/or before beamforming. However, the way of their transfer provides the processing only in post-processing mode. Our goal is to develop a universal platform that allows for the acquisition and real-time processing of raw echo signals on the GPU processors, and a practical implementation of the SAFT methods.

In recent years a number of research scanners were described in scientific literature. The most extensive is the SARUS system [1] offering support for up to 1024 channels of transmission and acquisition. The SARUS is based on the concept of hardware processing of the RF signal with very complex FPGA systems. It is a unique device because of its size, cost and complexity of the hardware implementation process. Another system developed by a research group is the ULAoP system [2] providing support for up to 64 channels of receivers. The FPGA systems are also applied in the ULAoP to perform the process of demodulation and beamforming whereas for further processing the DSP processor and the PC computer, which provides control and visualization, were used. An architecture with software processing of the RF signals (before the beamformer) was proposed in the commercial Verasonic system [3]. The Verasonic system supports up to 128 receive channels and is connected to the PC computer implementing the processing functions by use of the 8x PCIe interface. The entire processing and image reconstruction were carried out on the PC, equipped with two Intel® Xeon (6-cores) processors, using a patented "pixel-oriented" [4] algorithm.

The described in literature attempts to apply the GPU for ultrasound imaging are very promising but were limited to the implementation of software kernel processing, not a complete system. The limited bandwidth of the CPU↔GPU link was pointed out as a major obstacle for high frame rate imaging [5].

The proposed architecture of the acquisition system, the communication and the processing will ensure scalability of the system, both in terms of the number of acquisition channels as well as the available processing power. The use of PCIe switch fabric, and support for the direct peer-to-peer, using DMA (Direct Memory transfer) between the acquisition modules and the GPU cards, will eliminate the bottleneck of the CPU↔GPU memory transfer.

II. SYSTEM DESIGN

A. System Architecture

In classical ultrasound scanner architecture the raw RF data are used by a hardware beamformer (ASIC or FPGA) to create image scan-lines, which are further software processed. Multichannel RF data are concentrated and decimated (by beamforming) in the hardware processing section. It helps to limit output data bandwidth and processing requirements, while prevents any direct access or storage of the raw RF. Nowadays the GPU can handle massive amount of parallel processing making real-time implementation of ultrasound imaging feasible. However, depending on a number of the RF channels and complexity of applied imaging algorithm cumulative data bandwidth and/or processing can exceed single GPU PCIe performance.

The proposed system architecture (Fig. 1) is based on the standard PCIe switched fabric. By multiplying the number of acquisition modules and processing nodes one can scale both the number of acquisition channels and the processing power. Direct data transfers between the nodes (so called peer-to-peer transfers) are non-blocking on the switched PCIe fabric therefore they effectively multiply total system bandwidth.

Our ultrasound platform design consists of 1-3 pcs. of 64-channel acquisition module (RX64), single 192-channel transmission module (TX192), one control module (CTRL), and 1-3 pcs of GPU cards. Depending on configuration the system supports 64-192 acquisition channels.

All the system modules and processing GPUs are integrated in an external PCIe expansion chassis and connected to the main PC via PCIe gen 2 16-lane cable.

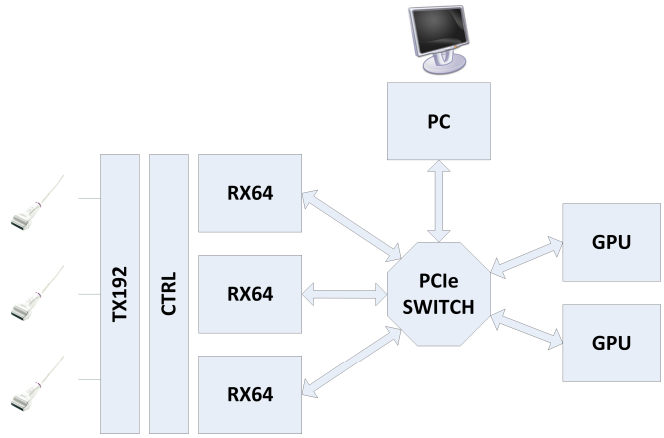

Figure 1. The proposed system architecture based on PCIe switched fabric.

B. The RX64 Acquisition Module

The platform is designed to be modular and scalable both in the number of ultrasound channels (64-256), as well as in the communication bandwidth and the processing power. By the usage of the standard PCIe switch fabric we plan to integrate all the ultrasound modules and the processing resources (GPU) in a single rack enclosure. The platform architecture consists of a PC, GPU cards and 3 types of custom electronic modules (acquisition, transmit, control). The acquisition module preforms digitalization of ultrasound echoes coming out from 64 transducer elements, data pre-processing and their streaming to the PC/GPU memory via PCIe interface for further processing. The number of acquisition modules is scalable depending on the number of required ultrasound channels. Application of the PCIe direct peer-to-peer communication for transfer of the data from the ultrasound acquisition modules to the GPUs maximizes the system bandwidth and minimizes the CPU usage. Additionally, direct memory transfers enable to multiply total system bandwidth in multi-module/multi-GPU configuration. The transmit module generates up to 192 excitation signals for each individual transducer element providing various different types (e.g. focused, plane) of the ultrasound wave to be generated. The control module generates synchronization signals and system clocks for the entire system.

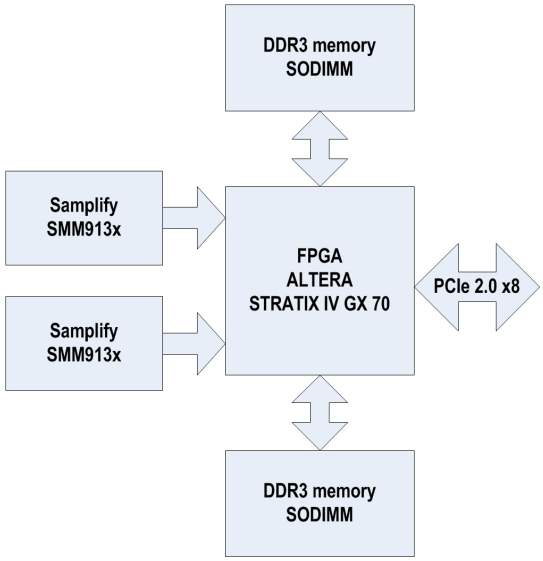

Figure 2. The RX64 module block diagram.

We designed and built a 64-channel acquisition module (RX64) (Fig. 2) equipped with 8 lane PCIe gen. 2 communication interface based on our platform architecture. The RX64 contains a high-end FPGA Stratix IV 70 GX (Altera, USA) interfaced to: two 32-channels mixed-signal front-end modules SMM913x (Samplify, USA) and two 64-bit 8 GB DDR3 SO-DIMM memories for data buffering. The Samplify modules provide analog conditioning of the ultrasound echoes, A/D conversion with 12-bit resolution at 65 MSPS, and data serialization. Serialized 64-channels data are transferred to the FPGA via 72 LVDS lanes (64 data lanes, 8 clock lanes) at 780 Mbps per data lane (49.94 Gbps total). This serial LVDS signaling solution reduces the number of required pcb traces and the amount of generated digital noise. Digitized 64-channels data after deserialization can be pre-processed in the FPGA (demodulation, filtration) and stored in the local DDR3 memory or streamed via PCIe to the PC/GPU. The FPGA offers both embedded transceivers and integrated hard PCIe IP controller which greatly eases the implementation of PCIe endpoint and saves logic resources and PCB real estate. The internal (64-ch ADC: 50 Gbps, DDR3: 1066 Gbps) and external (PCIe: 40 Gbps) interface bandwidth of the RX64 were balanced to meet real-time streaming requirements. The sophisticated clock distribution system, consisting of 10 clocks, generated by the TI CDCE70120 and external VCXO enables all the components of the system to work in a synchronous manner which is crucial to the proper operation of the whole system. The generated sampling clocks have superior jitter performance which minimizes the SNR of the acquisition system. The possibility of programming the phase shift of the sampling clocks allows to double the sampling frequency (130 MSPS) and reduce the number of channel to 32 by means of ADCs interleaving. The designed RX64 module PCB has 16 layers and dimensions of 264x164 mm. Sophisticated high speed digital design techniques, such as controlled impedance traces, differential signaling, length matching and crosstalk analysis, were used [6]. The power and signal integrity of the PCB was verified in the Mentor Graphics HyperLynx EDA software.

The first working prototype of the RX64 module (Fig. 3) was successfully launched. The PCIe interface and DDR3 memories were tested. The PCIe transfer speed tests showed throughput of 3 GB/s which is very close to the maximum achievable transfer rate of that interface [7]. The DDR3 memories run successfully at 500 MHz supporting up to 16 GB/s of memory throughput. The ADCs sampling quality was also investigated by performing FFT analysis on a pure, nearly full scale sine input [8] The results, presented in the table below, are very similar to the one specified in datasheet. The module dissipated power is 35 W.

TABLE I. ADC SAMPLING PARAMETERS

Parameter Name	*Measured Value*
SNR	67,1 dBFS
HD2	-60,9 dBc
HD3	-65,8 dBc
THD	-59,5 dBc

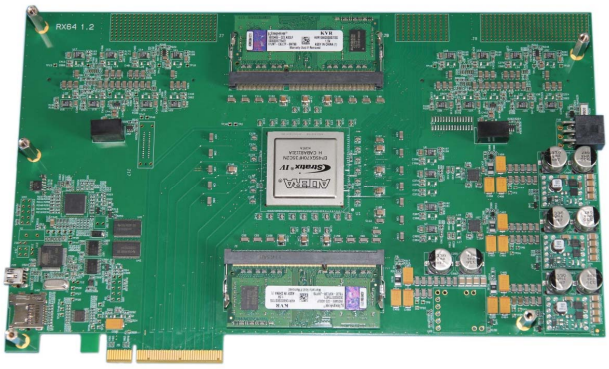

Figure 3. The RX64 module PCB view.

III. GPU PROCESSING

Our project goal is real-time acquisition, streaming and SAFT image reconstruction based on GPU computing. We started research and software development using Nvidia CUDA. We implemented the SAFT image reconstruction kernels to verify feasibility of real-time processing. On the other hand we developed a complete GPU Framework enabling running complete data flows using different GPU kernels. Since the SAFT algorithms are extremely computational and memory demanding we focused on kernel optimization for maximizing GPU resource usage.

A. GPU Framework

Beside hardware development we started software processing implementation being a vital part of the platform. To enable high level of programming and kernel customization we developed software GPU Framework. The Framework consists of an engine to run processing pipeline and an SDK (Software Development Kit) for implementing new processing kernels. Data processing flow has been abstracted as a stream defined by a graph (XML file). The graph is built as a combination of processing elements (nodes) and data connections (edges). Each node is composed of three software components: input strategy, GPU processing kernel and output strategy. The strategies are responsible for formatting the input and output data. The loaded and launched processing stream is run by the Framework – receiving the data from the acquisition cards and generating the outputs for the visualization application. The Framework supports the operations on many GPU cards, as well as different patterns for the data flows (eg. fork-join, scatter-gather).

Two SAFT imaging algorithms: STA (Synthetic Transmit Aperture) and PWI (Plane Wave Imaging) have been implemented, optimized and benchmarked on the GPU [9].

Our optimized SAFT implementation of Nvidia GTG-580 achieved frame rate of 8.5 Hz 512×512 pixel HRI (equivalent of 1100 LRI/s) for the non-decimated 128-channel IQ data of 55 mm depth. Reduction of the image resolution results in proportional increase of the frame-rate for the algorithm. For the output HRI at resolution 256×256 pixel the frame-rate equal to 31 Hz (4000 LRI/s) was obtained.

IV. CONCLUSIONS

The described RX64 acquisition module is the first element of the ultrasound universal platform being in a process of development. The current work includes designing of subsequent modules of the platform (the transmitter module - TX192 and the control module - CTRL), and integration with the GPU Framework which will provide software path for the ultrasound signal processing. In the next step the Framework will be extended with subsequent kernels for the ultrasound processing (including Doppler), as well as optimized for the new version 5 of the Nvidia CUDA.

The presented new ultrasound system architecture and the processing flow enable multichannel real-time ultrasound acquisition, streaming and GPU processing. The obtained GPU SAFT imaging performance proves the feasibility of the GPU computing for complex ultrasound algorithms. The hardware platform with GPU software Framework will hopefully become a tool for research, prototyping and development of new ultrasound methods.

ACKNOWLEDGMENT

Project POIG.01.03.01-14-012/08-00 co-financed by the European Regional Development Fund under the Innovative Economy Operational Programme.

INNOVATIVE ECONOMY
NATIONAL COHESION STRATEGY

EUROPEAN UNION
EUROPEAN REGIONAL DEVELOPMENT FUND

REFERENCES

[1] J.A. Jensen, H. Holten-Lund, R.T. Nielson, B.G. Tomov, M.B. Stuart, S.I. Nikolov, M. Hansen, and U.D. Larsen, "Performance of SARUS: A synthetic aperture real-time ultrasound system," Ultrasonics Symposium (IUS), 2010 IEEE, pp.305-309, 11-14 Oct. 2010.

[2] P. Tortoli, L. Bassi, E. Boni, A. Dallai, F. Guidi, and S. Ricci, "ULA-OP: an advanced open platform for ultrasound research,"Ultrasonics, Ferroelectrics and Frequency Control, IEEE Transactions on, vol.56, no.10, pp.2207-2216, 2009.

[3] http://www.verasonics.com

[4] R.E. Daigle, Ultrasound imaging system with pixel oriented processing, Patent: US 2009/0112095 A1, 2006.

[5] H.K.-H. So, J. Chen, B.Y.S. Yiu, and A.C.H. Yu, "Medical Ultrasound Imaging: To GPU or Not to GPU?," Micro, IEEE , vol.31, no.5, pp.54-65, Sept.-Oct. 2011.

[6] H. Johnson, M. Graham, *High-Speed Signal Propagation Advanced Black Magic*. Prentice Hall, 1993.

[7] R. Budruk, D. Anderson, T. Shanley, *PCI Express System Architecture*. Addison-Wesley Professional, 2003.

[8] W. Kester, *Data Conversion Handbook*. 3rd ed. Newnes, 2004.

[9] M. Lewandowski, P. Karwat, J. Kudelka, and T. Kleczek, "GPU Implementation of the STA Algorithm on I/Q Data," Ultrasonics Symposium (IUS), 2012 IEEE, unpublished, 7-10 Oct. 2012.