

Design and Evaluation of Reconfigurable Ultrasonic Testing System

Vidya Vasudevan, Boyang Wang, Pramod Govindan and Jafar Saniie

*Department of Electrical and Computer Engineering
Illinois Institute of Technology, Chicago, Illinois, U.S.A.*

Abstract – Ultrasonic data acquisition systems used in the medical imaging and nondestructive testing applications have critical requirements in order to be capable of evaluating and prototyping ultrasound applications. In this paper, a system-on-chip (SoC) based reconfigurable ultrasonic testing system (RUTS) is developed for implementation of high-speed ultrasonic signal processing algorithms. RUTS comprises of a fully reconfigurable analog front-end (AFE) sub-system for efficient data acquisition, and a Xilinx Zynq SoC module for dynamic feature control. Computationally intensive ultrasonic signal processing algorithms are implemented on Zynq SoC using hardware-software co-design methodologies. AFE supports up to 8 transducers for phased-array implementation. This Linux-based system is analyzed for ultrasonic data compression implementation providing a very versatile environment for further ultrasound system development and research work.

I. INTRODUCTION

Ultrasonic systems used in medical and material evaluation applications have a vital requirement for data processing capability. With limitations ranging from computational complexity to severe constraints in data/information transfer, there is a scope for advancement in the architecture-level implementation for such a complex system. This study focuses on one possible solution by leveraging the capability of programmable devices such as a System-on-Chip (SoC) as the back-end processor for computation and dynamic re-configuration of the system features [1]. The system built has demonstrated a capability to be remotely controlled, which provides an advantage for deploying it in areas or environment inaccessible at ease.

To demonstrate the capabilities of RUTS, an implementation based on a reconfigurable hardware [2] is built, and an example application of ultrasonic data compression is realized. The transducer interface is an analog excitation channel which provides for beamforming with 8 transducers connected to the system supporting operational frequencies in the range of 20KHz to 20MHz. The acoustic reflections termed as “echo” received by the system are filtered and amplified to provide

raw information to the processing sub-system for executing further ultrasonic signal-processing algorithms. The processing system used here is a Xilinx Zynq[®] SoC which integrates two ARM Cortex A9 processors running at 667 MHz, and a programmable logic fabric connected over ARM AMBA[®] AXI interconnect[3].

RUTS consists of a TX-SDK-V2 pulser/receiver board from Texas Instruments (TI) providing excitation pulses up to $\pm 50V$ on all 8 channels with configurable pulse patterns. This board is interfaced with a programmable filter/amplifier VCA8500Board from TI which provides up to 50dB gain. The amplified signal is digitized by a 16-bit 250MSPS Analog Devices (AD) AD9467 ADC [4]. The 16-bit data from the echo is buffered and send over a LVDS interface to Zynq for executing ultrasonic signal processing algorithms. The block diagram for the complete system is presented in Figure 1.

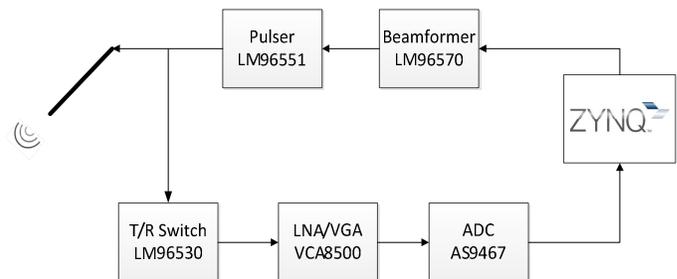


Figure 1. RUTS block diagram

In this paper, Section II provides a brief description of the Zynq SoC implementation. Section III explains the control loop programmed into the processing sub-system. Section IV presents the ultrasonic compression algorithm implemented on RUTS to analyze its performance. Section V concludes this paper.

II. ZYNQ SoC INTEGRATION

The RUTS is built on Xilinx Zynq 7020 SoC, which combines an ARM dual-core processing unit with an FPGA fabric. The main component of this system is the dual ARM Cortex A9 CPUs running at 667MHz with separate NEON

processors. Based on Harvard architecture, each processor has its own memory management unit (MMU) and 32KB level-one (L-1) instruction and data caches. The Cortex-A9 processor implements the ARMv7A architecture with full virtual memory support and can execute 32-bit ARM instructions. The advanced SIMD (single instruction multiple data) instructions help to execute the ultrasonic signal processing algorithms at a very high rate. One of the primary advantages of using the Zynq SoC is the availability of a programmable logic (PL) which significantly enhances system performance, reduces power consumption, and delivers predictable latency for real-time events. Hence, hardware-software (HW-SW) co-design strategy can provide multiple design solutions based on the application requirements and the algorithm to be implemented.

This work has many core considerations including the control logic built for the reconfigurable AFE features. The list of features is presented in Table 1. Each of these features are controlled by the Zynq SoC via SPI (serial peripheral interface) communication channel to program each of the devices which includes beam former, pulser, transmit/receive (TR) switch, and, amplifier [5]. Zynq supports a range of peripherals to communicate with other devices including two full-duplex SPI controllers with three peripheral chip selects. This port is used to interface with the other components of the transducer interface sub-system for dynamically writing into and therefore, programming the devices to a specific requirement. The integration of various components within Zynq SoC is shown in Figure 2.

Table 1. RUTS reconfigurable features

Feature	Value Range
Channel(s)	1 to 8
Pulse Length	4 to 64 pulses
Operating Frequency	625KHz to 80MHz
Channel Delay	0.78ns to 102.4 μ s
Echo Gain	40dB to 50dB
Sampling Frequency	250 MSPS
Transient Options	50V/600mA or 50V/2A

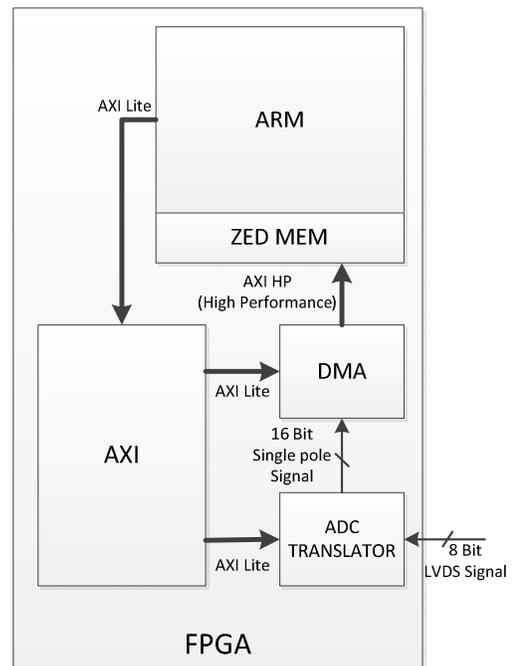


Figure 2. Integration of components within Zynq SoC

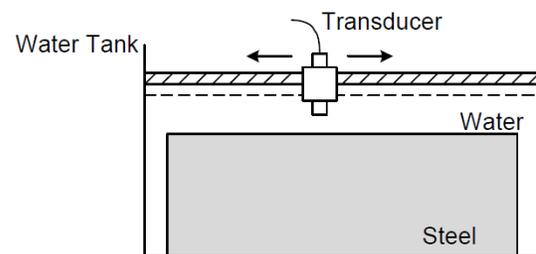


Figure 3. Ultrasonic Test Setup

Figure 3 illustrates the test setup used to evaluate RUTS. This setup has a motor assembly which moves the transducer in X and Y directions to collect multiple A-scans. The motor and its movement are precisely controlled by the Zynq. The movement of the transducer across the face (B-scan) of the target steel block test specimen (top view) is shown in Figure 4.

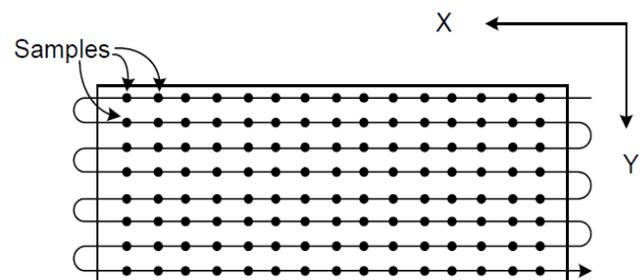


Figure 4. Ultrasonic B-scan scanning pattern

III. ULTRASONIC DATA COMPRESSION

The high speed ADC in the system samples the ultrasonic data at 250MSPS with each sample having 16-bits. This results in a throughput of 500MBPS. With 8 channels enabled, the throughput significantly increases to 4GBPS on a serial LVDS output from ADC. For such a huge amount of data, an efficient compression method needs to be implemented on the Zynq system to compress the digitized data. The compressed data can be stored in the on-board memory or it can also be transmitted to other platform by using communication methods such as Ethernet or serial port. Furthermore, the data can be wirelessly transmitted over the internet to a remote server. The data from the remote servers can be de-compressed at a later point in time for further processing.

Since the ultrasonic signal is a non-stationary signal, Discrete Wavelet Transform (DWT) is used to compress the echo signal. Figure 5 shows the block diagram of the DWT compression implementation [6].

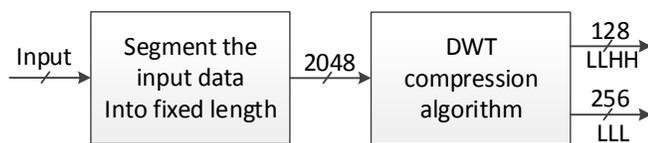


Figure 5. DWT compression block diagram

After the digitized time-domain signal is input to the system, Zynq will start processing the data. Initially, the input data will be segmented and buffered into small sections of defined length; which is a power of 2. Here, a length of 2048 samples is chosen. A four level of DWT is applied to the buffered data to generate narrower sub-bands. The basic structure of each level of the DWT compression is shown in Figure 6. The original signal is input to a low pass sub-filter and a high pass sub-filter. The output of each sub-filter is down-sampled by two. Certain sub-bands with very low energy are eliminated to achieve compression [6].

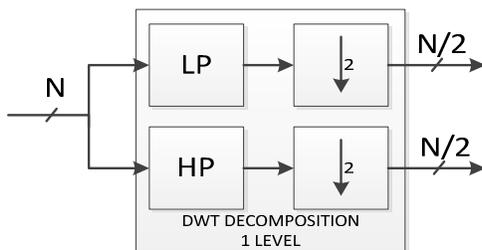


Figure 6. Structure of single level discrete wavelet transform

Figure 7 shows a portion of signal captured by AD9467 with a Zynq SoC. This signal represents reverberation echoes acquired from a layered steel test specimen. The signal is sampled at 250MSPS resulting in 65,535 samples that have to be compressed. Figure 8 shows the compressed signal. After carefully analyzing the results in Matlab, the two sub-bands, namely LLL (256 samples) and LLHH (128 samples) are selected to represent the overall data. Thus, a compression ratio of 81% is achieved. The signal at the top of Figure 8 is the LLL sub-band, and the signal at the bottom is the LLHH sub-band.

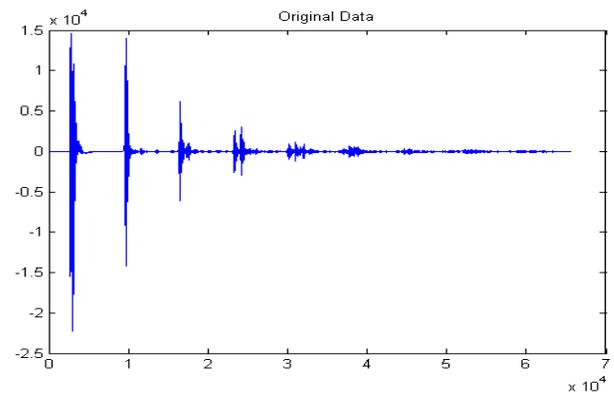


Figure 7. Measured ultrasonic signal from a reverberant test specimen

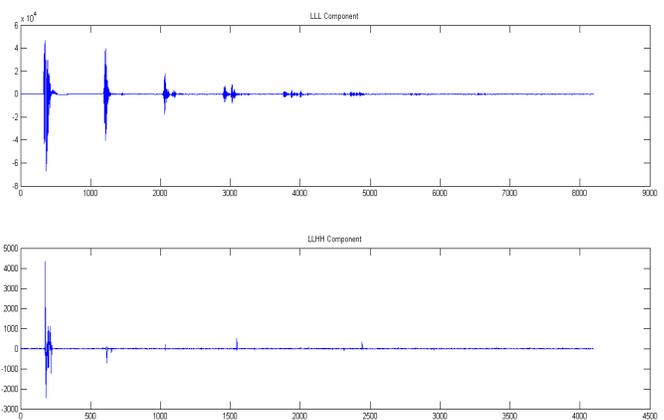


Figure 8. Compressed signal

Figure 9 shows the reconstructed signal. It can be observed that the reconstructed signal exactly follows the original signal shown in Figure 7. Furthermore, the correlation of the reconstructed signal and the original signal is 99.99%, which shows very high signal fidelity.

The ultrasonic data compression algorithm is executed on different platforms to compare the performance. Table 2 lists the execution time for compressing 65,535 samples of acquired ultrasonic data on three different platforms.

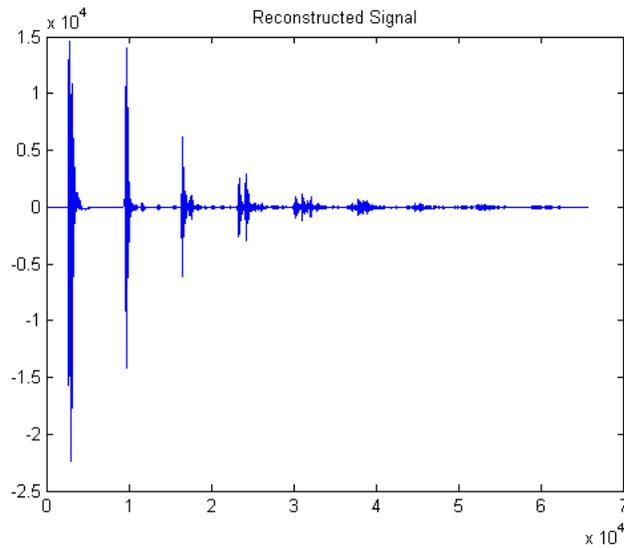


Figure 9. Reconstructed signal

Table 2. Execution time of ultrasonic signal compression on different platforms

Time (s)	Intel i7	Xilinx Zynq	Raspberry Pi
Decomposition	0.003	0.03	0.22
Reconstruction	0.005	0.07	0.87
Total	0.008	0.1	1.09

The target platforms used in this study are the Intel i7 personal computer, Xilinx Zynq based Zedboard and Raspberry Pi board. The data input speed is decided by ADC which is 250 Mbps. Every second, there will be 250Mb of samples being input to the system, considering the ADC is capturing data at the highest sampling rate. To compress the data in real time, the compression time for 65,535 samples of data need to be less than 2.6×10^{-4} s which is impossible for the ARM processor on Zedboard or Raspberry Pi board.

There are two solutions to this issue. The first solution is being used in this study, which is to buffer the target data in the memory and gradually compress all the data. In this case, the processor speed is not a limitation since the processor signals a waiting sequence to the ADC for pausing data capture until the data compression is finished. However, this is limited by the buffer memory requirements.

The second solution is to implement the DWT compression algorithm on the Programmable Logic (FPGA) within the Zynq SoC. The high speed data acquired by ADC will be compressed by the hardware before it is processed by the processor. In this way, a real time signal processing system can be realized.

IV. CONCLUSION

RUTS provides a versatile environment for executing computationally intensive ultrasonic signal processing algorithms by enabling dynamic control over the various aspects of transducer excitation to suit different application requirements. Furthermore, Zynq SoC within RUTS allows flexible implementation options such as hardware software co-design, by using programmable logic resources along with capabilities of powerful ARM processor. While providing a more portable solution to challenges faced in the NDT industry, highly capable equipment presented in this work is proven to support multitude of applications.

REFERENCES

- [1] A. Kassem, M. Hamad, M. Sawan, "An Efficient SoC Dedicated to Ultrasonic Digital Imaging Systems", *The 6th International Workshop on System-on-Chip for Real-Time Applications*, pp.165 - 168, December 2006.
- [2] V. Vasudevan, P. Govindan, J. Saniie, "Reconfigurable Analog Front-End System for Ultrasonic SoC Hardware", *IEEE International Conference on Electro/Information Technology (EIT)*, pp. 356 – 361, June 2014.
- [3] Xilinx, "Zynq-7000 All Programmable SoC Overview", DS190 (v1.7) Oct 8, 2014 [Online]. Available: http://www.xilinx.com/support/documentation/data_sheets/ds190-Zynq-7000-Overview.pdf
- [4] Analog Devices, "AD9467 Native FMC Card / Xilinx Reference Design", Feb 9 2015 [Online], Available: http://wiki.analog.com/resources/tools-software/linux-software/zynq_images
- [5] V. Vasudevan, P. Govindan, J. Saniie, "Dynamically Reconfigurable Analog Front-End for Ultrasonic Imaging Applications", *Ultrasonics Symposium (IUS), 2014 IEEE International*, pp. 1924 - 1927, September 2014.
- [6] P. Govindan, J. Saniie, "Processing Algorithms for Three-dimensional Data Compression of Ultrasonic Radio Frequency Signals", *IET Signal Processing Journal*, 2015.