Optimization of Vertical and Horizontal Beamforming Kernels on the PowerPC G4 Processor with AltiVec Technology

EE382C: Embedded Software Systems
Literature Survey

David Brunke
Young Cho
Applied Research Laboratories:
The University of Texas at Austin

Abstract

Real-time digital sonar beamforming is a computationally intensive algorithm that has been implemented in the past primarily in custom embedded hardware. With recent advancements in native signal processing extensions for general-purpose processors, it is possible to implement sonar beamforming using off-the-shelf hardware. Current implementation on a Sun UltraSPARC multiprocessor suggests a very promising platform for transitioning such applications to general-purpose systems. This paper proposes to continue the previous implementation by modifying the beamforming kernels to use AltiVec, a new native signal processing extension from PowerPC. AltiVec is a Single Instruction Multiple Data (SIMD) architecture capable of executing up to four 32-bit floating-point multiply and accumulate (MAC) operations per instruction. The kernels utilizing such powerful features of the AltiVec are expected to perform with significant speedup over previous implementations on other general-purpose processors.

1.0 Introduction

In last few decades, the Digital Signal Processor (DSP) market has grown substantially to meet the demand of the high performance signal processing community. Such growth in the signal processing market has allowed the general computing communities to incorporate the technology into their applications. As a result, general-purpose processors began to embed signal processing architectures into their own processing cores to provide economic system solutions for computationally intensive applications [1].

Real-time digital sonar beamforming is one such application once only feasible on custom hardware that can now be successfully implemented on commercial, off-the-shelf computers with native signal processing extensions. One recent implementation uses a commercial general-purpose 8-way symmetric multiprocessor (SMP) workstation from Sun Microsystems [2]. The beamforming kernels exploit the inherent data parallelism by using Single Instruction Multiple Data (SIMD) arithmetic operations available in the Visual Instruction Set (VIS) extensions to the UltraSPARC processor. By using a sixteen 333-MHz UltraSPARC Enterprise server, a real-time beamformer delivering 4 GFLOPS on 160 MB/s of streaming data was realized.

The goal of our research is to further explore the effectiveness of the embedded extensions by optimizing and assessing the performance of beamforming kernels using AltiVec from PowerPC, which is one of the newest native signal processing instruction sets. We also plan to analyze the results obtained from the two embedded signal processing extensions to assess the architectural advantages and disadvantages.

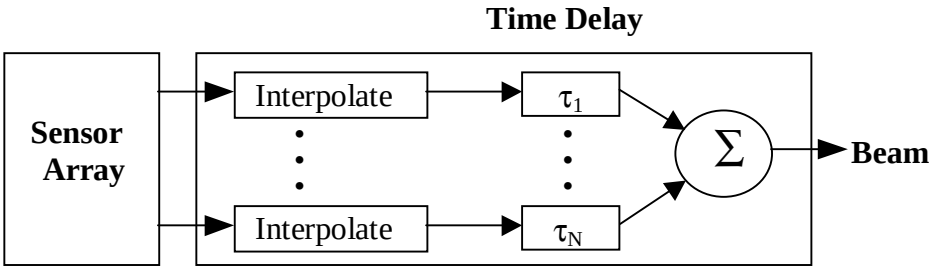

Fig. 1: Digital Interpolation Beamformer

2.0 Beamforming Approach

Conventional sonar beamformers use the signals collected from sensor elements to determine from what direction the sonar signal returns after deflecting off of an object. This conventional horizontal time-domain beamforming algorithm consists of appropriately delaying and summing the weighted outputs of an array of sensor elements. The weighting of the sensor outputs helps to improve the spatial response [3].

The problem with this conventional approach is that it requires a sample rate that is several times the Nyquist rate for adequate time delay resolution. This is undesirable because it requires additional bandwidth for the overall system. A practical solution employs digital interpolation with Finite Impulse Response (FIR) interpolation filters to achieve a satisfactory time delay resolution [3]. This solution is shown in Fig. 1. Analog data is sampled at a given sampling interval, and then followed by interpolation, time delay, and summation.

3.0 Native Signal Processing Extensions

Many high performance embedded applications are programmed on systems with a few general-purpose processors as system controllers with a larger number (possibly hundreds) of specialized DSPs to perform scientific calculations. However, this type of system has many disadvantages due to different programming platforms and unequal

performance advances in the two separate technologies. Therefore, many manufacturers of high performance general-purpose processors are integrating sets of native signal processing instructions onto their processor cores to offer solutions requiring fewer processors.

3.1 UltraSPARC VIS

The Visual Instruction Set (VIS) is a set of signal processing instructions based on the SIMD architecture. The floating-point data of the UltraSPARC processor core is enhanced with graphics units to support VIS. Although graphics units share the register file resource with the floating-point units, they are distinguished from floating-point units by performing fixed-point vector arithmetic. VIS provides over 50 new CPU instructions such as format conversions, arithmetic and logic instructions, address handling, memory access instructions, and several others. Equipped with 64-bit registers, they can be partitioned with 2, 4, or 8 data words, and can perform operations on multiple words with a single instruction. Thus, VIS can achieve up to four times speedup with 8-bit by 16-bit fixed-point multiplication using the SIMD arithmetic logic [4].

3.2 PowerPC AltiVec

AltiVec is the native signal processing extension for PowerPC processors. This short vector SIMD architecture is embedded into a general purpose RISC processor core to add powerful signal processing capabilities. Unlike the UltraSPARC VIS, AltiVec vector logic is not an enhancement to an existing arithmetic unit. It is sectioned into a separate sub-unit of the processor as are the floating-point and integer units. The vector unit adds more than 150 new SIMD instructions to the PowerPC instruction set for advanced signal processing programming.

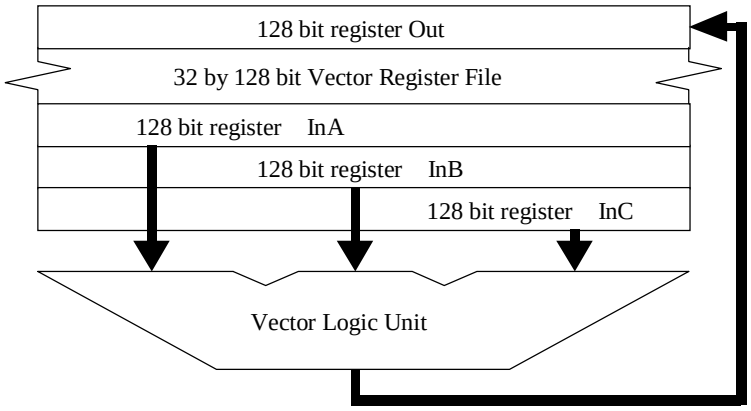

Figure 2: Block Diagram of PowerPC AltiVec Unit Architecture

The vector unit has its own 32 by 128-bit wide register file that allows execution of up to four 32-bit floating point MAC operations per instruction [5]. AltiVec is potentially a much more powerful signal processing extension than VIS due to its greater logic resources.

4.0 Current Implementation

The current implementation that we are going to build upon adds vertical beamforming to the approach in [3] to enable projection of a 3-D underwater image [2]. In addition, the interpolation for the horizontal beamforming kernel is simplified by using a two-point FIR filter for the digital interpolation. A two-point FIR filter is possible without a critical loss of resolution because the sampling rate is set to two times the Nyquist sampling rate. The overall system description is shown in Fig. 3.

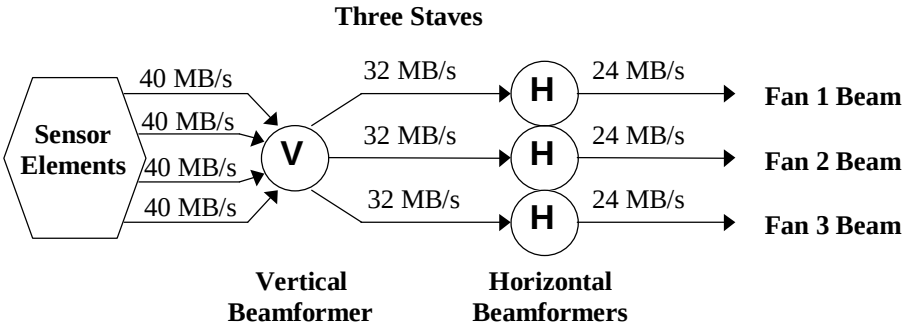

Figure 3: Block Diagram of the 3-D sonar beamformer

4.1 Vertical Beamforming

The vertical beamformer computes three sets of data, each of which is sent to a horizontal beamformer. Thus, three dot products are computed with each column of 10 vertical transducers (staves) and three coefficient vectors; each contributes to the vertical resolution. In the current implementation, a given sample for 80 horizontal elements with 10 vertical transducers requires 2400 MACs. After the vertical beamforming, the output needs to be in floating-point format for the horizontal beamformer. Therefore, integer-to-floating-point type conversion needs to be performed on the result [2].

4.2 Horizontal Beamforming

In the current implementation, the time delays can be determined by projecting the semi-circular array of sensor elements onto an axis perpendicular to the pointing direction for each beam. The time delay is then defined as the distance from each element to the perpendicular axis divided by the speed of sound. Not all of the elements are used to compute the desired beam because the response in their directions is relatively small, causing unnecessary calculation [6].

A two-point FIR filter is used to perform the time delay interpolation to achieve the desired horizontal steering delay resolution. A total of 61 beams are formed from 80 elements, with 50 horizontal elements used per beam. Each beam sample, $b_i[k]$, is made up of a two-point FIR filter for each sensor element x_n, which weights, delays, and adds two time samples. The weightings, w_{kn0} and w_{kn1}, are determined by the delay fraction and beamformer shading. The delay, τ_{in}, for each beam i and sensor n, is an integer sample delay. Thus, the beam sample output $b_i[k]$ is given by the following equation, where each

beam sample requires approximately 50 index lookups and 100 MACs. So approximately 3000 index lookups and 6100 MACs must be performed for all 61 beams [6].

$$b_i[k] = \sum_{n=1}^{50} w_{kn0}\, x_n[k-\tau_{in}] + w_{kn1}\, x_n[k-\tau_{in}-1]$$

4.3 Benchmark Results

For the current implementation using VIS instructions, the UltraSPARC processor executes one signal processing operation per cycle. The vertical beamformer requires 4800 operations per sample. Benchmarks for the vertical beamforming kernels show a requirement of 1.3 billion operations [2]. The performance results for VIS versus non-VIS code for the vertical beamforming kernels is shown in Fig. 4. The VIS code almost triples the number of operations given by the non-VIS code.

The horizontal beamformer consists of digital interpolation with single-precision (32-bit) floating-point numbers. Each sample requires over 12,000 single-precision floating-point operations and 6,000 index lookups. According to the benchmark results, the horizontal beamforming kernels require a total of 3.2 billion floating-point operations [2].

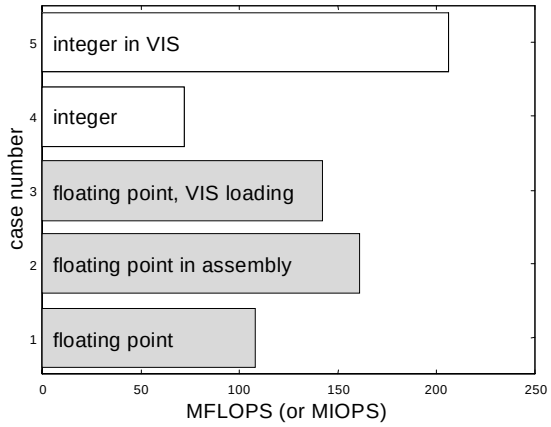

Fig. 4: Performance Results for the Vertical Beamforming Kernels

5.0 Proposed Implementation

With recent releases of G4 PowerPC processors with AltiVec and the AltiVec enabled C compiler, we can confirm the evaluations assessed with various simulators [7]. We can evaluate the results of the vertical and horizontal beamforming kernels programmed with AltiVec on the G4 processor in a real working environment. In the future, we will be able to evaluate the performance of AltiVec in a PowerPC G4 four-way SMP system.

5.1 Beamforming Kernel

Due to the structural differences between VIS and AltiVec instruction architectures, the input data and the program layout needs to be changed to fully utilize the AltiVec extensions for the kernels [5,8,9]. Since AltiVec provides a four-way 32-bit SIMD architecture, the vertical kernel using AltiVec can optimally yield four MACs per cycle to speed up the calculation. Four 32-bit floating-point SIMD instructions can be utilized to exploit its parallelism for the horizontal beamforming kernel as well. In terms of instruction cycles, such modifications can potentially give speedup of six to ten times the previous implementation.

6.0 Conclusion

This project evaluates the path that general-purpose processor manufacturers have taken to accelerate the signal processing algorithms which are increasingly becoming popular not only in scientific computing but in personal computing as well. Advances in embedded signal processing technology within general-purpose processors introduce powerful and economic solution even for algorithms that used to require custom hardware. This project proposes to benchmark one such computationally intensive algorithm, sonar

beamforming, on two different native signal-processing technologies. The kernel implementation and its benchmark will be analyzed to reveal the advantages and disadvantages of architectural integration [10].

7.0 References

[1] J. Bier, "DSP on General Purpose Processors," *MicroDesign Resources Dinner Meeting Slides, Berkeley Design Technology, Inc.*, Jan. 1997.

[2] G. E. Allen and B. L. Evans, "Real-Time Sonar Beamforming on Workstations Using Process Networks and POSIX Threads." *IEEE Trans. on Signal Processing*, vol. 48, no. 3, pp. 921-926, March 2000.

[3] R. G. Pridham and R. A. Mucci, "A Novel Approach to Digital Beamforming." *Journal Acoustical Society of America*, vol. 63, no. 2, pp. 425-434, Feb. 1978.

[4] Sun Microsystems, "VIS Instruction Set User's Manual." *http://solutions.sun.com/embedded/databook/pdf/manuals/805-1394-01.pdf.*

[5] AltiVec Programming Environment/Interface Manual, *Motorola Inc*., 1998.

[6] G. E. Allen, B. L. Evans, and L. K. John, "Real-Time High-Throughput Sonar Beamforming Kernels Using Native Signal Processing and Memory Latency Hiding Techniques", *Proc. IEEE Asilomar Conf. on Signals, Systems, and Computers*, Oct. 25-28, 1999, vol. I, pp. 137-141, Pacific Grove, CA.

[7] H. Nguyen and L. K. John, "Exploiting SIMD parallelism in DSP and multimedia algorithms using the AltiVec technology." *Proc. ACM Int. Conf. on Supercomputing*, June 20 - 25, 1999, pp. 11-20, Rhodes Greece.

[8] Motorola, "AltiVec Technology." *http://www.mot.com/AltiVec.*

[9] AltiVec Information Source. *http://www.altivec.org.*

[10] J. D. Allen and D. E. Schimmel, "Issues in the Design of High Performance SIMD Architectures." *IEEE Trans. on Parallel and Distributed Systems*, vol. 7, pp. 828-839, Aug. 1996.