Exploring Multi-core Processors using Realistic Signal- and Image-processing Application Benchmarks.

Ray E. Artz, Brian J. Loe, and Janet M. Pavelich, {ray.e.artz, brian.j.loe, janet.pavelich}@lmco.com
Lockheed Martin MS2 Tactical Systems (Eagan, MN)
Jules P. Bergmann, jules@codesourcery.com
CodeSourcery, Inc.

Overview

An ongoing study is being conducted to benchmark performance of two multi-core processors, namely the IBM Cell[1] and the P.A. Semi PWRficient PA6T-1682M[2]. These will be benchmarked using three realistic applications, namely (1) the HPEC Challenge SAR benchmark[3], (2) a Lockheed Martin electro-optical (EO) application, and (3) a Lockheed Martin atmospheric- acoustic application which includes a minimum-variance distortionless-response adaptive beamformer. These three applications will be coded using (a) CodeSourcery VSIPL++[4] and (b) Mercury Multi-core Framework (MCF)[3] and Scientific Application Library (SAL)[5] Application Programming Interfaces (API's) and libraries. Application performances will be measured on both multi-core processor types under a variety of clock-speeds and operating conditions.

The HPEC 2007 presentation will present progress to date; as such, it will focus on the execution of the EO and SAR applications on the PA6T-1682M processor.

Background

For the past several years, there has been an apparent slow-down in the so-called "Moore's Law" in the sense that it has not been possible for embedded computer systems to continue to get significant increases in computing performance each year unless the programs were also prepared to accept significant increases in power dissipation. Also, real throughput has increased more slowly than CPU clock speed for various reasons (memory-speeds, pipelines, cache design, etc.) In one unnamed DoD program of interest, over a 5-year period, processor clock speeds were increased from 500 MHz to 1.2 GHz, but power per processor also increased (roughly in proportion to clock speed: 5.3W to 11.5W) and achieved throughput on real applications improved less than 60%. A new generation of multi-core processors offers promise of reversing this disappointing lag in performance compared to power dissipation. Although their characteristics are quite different, both the IBM Cell and the P.A. Semi PWRficient processor families offer, at least in theory, excellent processing performance per watt. (Of the two, the IBM Cell offers better potential performance per watt, but at the cost of a much higher-wattage individual package and a more complicated programming model.) The current study will investigate the performance of these processors using realistic benchmark applications implemented with several different software API's and performance-enhancing libraries.

Development platforms

The development platform for the IBM Cell Processor will be the Mercury Cell Workstation Development System[6]. The development platform for the P.A. PWRficient processor will be PWRficient Evaluation Kit[7] with the A.2 silicon version of the PA6T-1682M processor. Both platforms include various software development tools and libraries. In addition, we have obtained VSIPL++3.2 from CodeSourcery and expect to obtain additional software toolsets and libraries from P.A. Semi, Mercury Computer, and CodeSourcery as their software becomes available.

Benchmark application software

Three applications will be used as benchmarks in this study:

1. The HPEC Challenge SAR benchmark. This benchmark is available in MATLAB code and also (from CodeSourcery) in VSIPL++. Only the computational portion (in particular, excluding I/O) will be used in this study. A version of the benchmark will be developed using Mercury proprietary API's (MDF, SAL) for use in the study.

2. A Lockheed Martin electro-optical detection and tracking application. This benchmark is currently coded making use of the Mercury SAL library. In the course of the study, both a VSIPL++ version and a Mercury MCF version (for the IBM Cell) will be developed. One key feature of this benchmark is that performance data has already been collected for a number of processors, including MPC 7410, 7447, 7447A, and 7448, at various processor clock and memory speeds. Also, this benchmark poses challenges for implementation on the cell processor, because the Cell-processor 256-kByte SPE local-store memories are not big enough to hold an entire image. Hence, the application will have to be parallelized to be executed on the Cell processor.

3. A Lockheed Martin atmospheric- (as opposed to underwater-) acoustic application. This application includes a variety of signal-processing algorithms including an adaptive beamformer. The application is currently coded in MATLAB, with some functions implemented in FORTRAN. In the course of the study, both VSIPL++ and Mercury-proprietary-API versions of the application will be developed.

Software API's and methodologies

Three software API's will be of interest in this project:

1. MATLAB. No MATLAB code will be executed on target hardware for this study, but two of the three benchmarks exist in MATLAB code as a "starting point" for real-time code development.

2. Mercury-proprietary API(s). The electro-optical benchmark was originally implemented using the Mercury SAL library. Also, the Mercury proprietary API's are generally implemented in highly optimized code which can yield very good performance.

3. VSIPL++. This API offers code portability, a promise of high software productivity, and, at least in some cases, high performance. One way that CodeSourcery VSIPL++ can achieve high performance is by linking to optimized libraries (including, but not limited to, Mercury proprietary libraries) "under the hood", that is, while maintaining the VSIPL++ API at the application level.

The study will, to the greatest extent feasible, study the performance of CodeSourcery VSIPL++ both with and without Mercury-proprietary libraries "under the hood".

Study issues and methodology

The core of the study effort will involve making timing measurements for various benchmark software implementations (based on VSIPL++, Mercury Proprietary API, etc.) on the two multi-core processors of interest under a variety of conditions. Besides gross performance, issues of interest include:

1. Performance under component-power constraints. Some of our DoD programs of interest require processing to performed under constraints on power dissipation per-component; typical constraints could be maximal power dissipation of 6W-20W per component. For these programs it is not enough to achieve high throughput per watt; the wattage of each component must be kept low. It is clear that the IBM Cell processors are not candidates for such programs, but the P.A. Semi PWRficient processors could be, especially if the processors were operated at appropriate clock speeds. Performance does not necessarily scale linearly with CPU speed (especially when memory operations are involved), so benchmarks on the PWRficient processors will be carried out at a variety of processor and memory speeds.

2. Software-development productivity. It is well-known that software development costs are an ever-increasing component of overall DoD program costs. The current study cannot make a comprehensive unbiased assessment of software productivity using the VSIPL++ and Mercury-proprietary API's. There are too many confounding variables for unbiased investigation: experience and training of the software developers, quality standards for the software (laboratory use vs. customer use, etc., etc.). Nonetheless, engineering logs will be maintained regarding development times required to convert the various benchmark applications into code using the two API's for the two processor families of interest, and summaries of this data will be reported.

Limitations

Benchmark studies using recently-released hardware and software components necessarily suffer from some limitations. Some of the limitations of the current study are:

1. Hardware immaturity: The A.2 silicon in the P.A. development kit operates at neither the full speed nor the full architectural efficiency of the production components planned for release later this year.

2. Software library immaturity. It is very unlikely that either CodeSourcery VSIPL++ or Mercury libraries will be fully optimized for the multi-core processors being studied during this phase of the study.

3. Inter-processor communications issues. The development systems being used will not permit the study to seriously examine issues of inter-processor communications. The only inter-processor communications which will play a role in this study will be the communications among the multiple SIMD Processing Elements (SPE's) internal to a single Cell processor.

4. Proprietary data issues. Some of the benchmarks are based on Lockheed Martin proprietary application software. In these cases, it is expected that summary performance data (at the application level) will be presented, but detailed performance (which would disclose the details of the algorithms) will not be available for presentation.

Acknowledgements

The authors have received significant technical support from Mercury Computer Systems, Inc., P.A. Semi, Inc., and numerous members of the HPEC community in the course of this project, and would like to extend their heartfelt thanks.

References

[1] http://researchweb.watson.ibm.com/cell/, http://www.mc.com/microsites/cell/

[2] http://www.pasemi.com/

[3] http://www.ll.mit.edu/HPECchallenge

[4] http://www.codesourcery.com/vsiplplusplus

[5] http://www.mc.com

[6] http://www.mc.com/products/systems.aspx

[7] http://www.pasemi.com/processors/pwr_evaluation_kit.html