Algorithm Design Methodology for Embedded Architectures*

Kiran Kumar Matam[1] and Viktor K. Prasanna[2]

[1] Computer Science Department
[2] Ming Hsieh Department of Electrical Engineering
University of Southern California

Abstract. Power efficiency is a critical constraint for embedded systems. To address this many technological innovations are being proposed by the community. However, leveraging such advances also requires algorithmic solutions to handle the potential for run-time errors due to near threshold computing. In this work we plan to develop algorithmic innovations and optimizations for power efficiency in the emerging landscape of embedding computing platforms. We also plan to develop resilient run-time systems and incorporate resiliency in algorithmic solutions. In this paper we briefly describe our design methodology employed in the TAPAS (Tunable Algorithms for PERFECT Architectures) project. The TAPAS project is funded under the DARPA PERFECT (Power Efficiency Revolution for Embedded Computing Technologies) program.

1 Introduction

Many technological advances are being proposed by the community, which must be effectively exploited at the software and application layers to achieve and sustain the power envelope of a given application. Optimization at the algorithmic level has a much higher impact on total energy dissipation than microarchitecture or circuit level. Some recent studies have shown that the impact ratio is 20:2.5:1 for algorithmic, register, and circuit level energy optimizations. In this work we plan to develop algorithmic innovations and optimizations for power efficiency in the emerging landscape of embedding computing platforms. We plan to achieve the following objectives :

- Develop algorithmic optimizations for latency, throughput and energy performance,
- Identify opportunities ("knobs") to integrate these into the compilation capability and also enable overall application composition, and
- Demonstrate improved energy and resilience performance for signal processing kernels on next generation embedded computing platforms.

* This work has been funded by DARPA under grant number HR0011-12-2-0023.

P. Brisk, J.G. de Figueiredo Coutinho, P.C. Diniz (Eds.): ARC 2013, LNCS 7806, pp. 231–232, 2013.

2 Technical Approach

We plan to develop model-based algorithmic techniques [1] in which the design space can be explored for a given target embedded platform by considering various novel design time optimizations, coarse performance modeling and hierarchical design space exploration.

- **Tunability:** Our parallel algorithms for kernels will be specified using a small number of parameters including latency, energy, resiliency, input (problem) size, and number of processors. The methodology will permit the algorithm space to be explored by the designer by varying the parameters. Thus, we obtain tunable performance by varying these parameters and the algorithm execution specifies a surface in the three dimensional space of execution latency, energy consumed and achieved resiliency.
- **High level coarse performance modeling:** The space of embedded platforms is rather large. We do not expect one model to capture the key features of all target platforms or the features of a specific target platform to sufficient accuracy to explore the performance tradeoffs when mapping to that platform. Rather, we define a high level performance model called Integrated Computational Model (ICOM) which will be customized for each architecture-algorithm pair and a target platform. The model is static in the sense that the parameters are known and are estimated at design time. Also, it is a high level (coarse grained) model, by which we mean few parameters are used to get a coarse estimate of performance.
- **Exploring algorithm design space:** For a given kernel, many parallel techniques can be explored. Several parallel implementations for Discrete Fourier Transform (DFT) such as radix-2, radix-4, decimation in time or frequency can be specified as an architecture-algorithm pair. In our design methodology, each pair is explored for possible mappings onto the target platform by using ICOM. Thus, given a problem size and target platform resources, we explore the design space and generate a set of design choices for each value of latency, energy and resiliency. Detailed implementation or simulation is performed for these designs to choose an energy optimal one. We generate a surface of feasible solutions by varying performance parameters: latency, energy and resiliency.
- **Exploring various target architectures:** Our methodology is intended to model various target architecture features and explore them at the software layer. The features to be modeled can also depend on the characteristics of the architecture-algorithm pair. Thus, we will not develop a single model to capture all target embedded platforms or space of parallelizations, but rather use a customized model specific to a target embedded platform with parameters identified for that platform and the architecture-algorithm pair.

Reference

1. V. K. Prasanna, "Energy-Efficient Computations on FPGAs," *J. Supercomput.*, vol. 32, no. 2, pp. 139–162, May 2005.